CHIP

Auf DVD

Bootfähige DVD: Das RasPi-System für jeden PC
Plus: Tools, Codes und Extras zu den Workshops

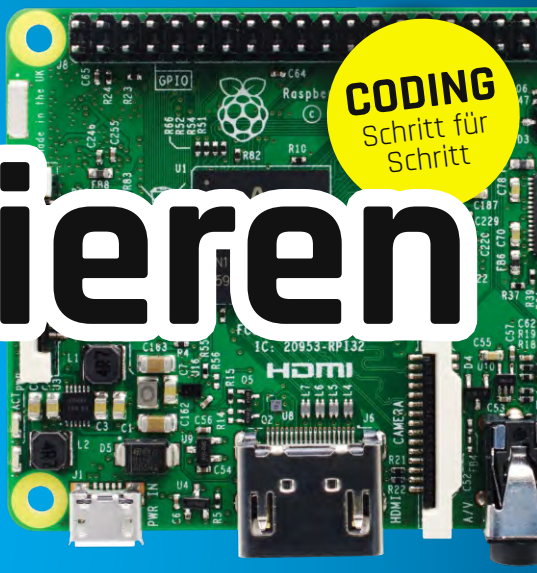
Das offizielle RASPBERRY PI Magazin

www.magpi.de

MagPi

02 • 2017
MÄRZ/APRIL

Das offizielle **Raspberry Pi** Magazin



CODING Schritt für Schritt

Einsteigerkurs Programmieren

Coding leicht gemacht: Entdecken Sie völlig neue Möglichkeiten mit Ihrem Raspberry Pi

GPIO-GRUNDLAGEN

Verständlich erklärt: So nutzen und programmieren Sie die GPIO-Pins

GOOGLE DEEPDREAM

Wenn Computer träumen: Surreale Bilder mit Ihrem Raspberry Pi

MEDIACENTER

LibreELEC: Was bringt das neue Multimedia-System wirklich?

LIVE-SYSTEM AUF DVD
Pixel für den PC

Das RasPi-System jetzt auch für Computer

Workshops:

- Ein Roboter, der Gefühle erkennt
- Überwachungskamera mit Sensor
- Musikbox: Sound auf Knopfdruck

DT-Control geprüft:
Beiliegender Datenträger ist nicht jugendbeeinträchtigend

02 • 2017 • € 9,95
ÖSTERREICH: 11,50 EUR BENELUX: 11,50 EUR
SCHWEIZ: 19,50 CHF

4 198283 109957 02

IM TEST: BOARDS, ROVER-KIT, NEXTCLOUD-BOX

CHIP Academy

VIDEO-KURS

Lightroom perfekt im Griff

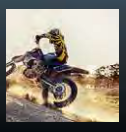
HEICO NEUMEYER

6 KAPITEL 23 FILME 210 MINUTEN 19,90 €

Neu!
CHIP Academy
Video-Tutorials rund um Technik, Foto, Internet und Business

CHIP Academy-Dozent **HEICO NEUMEYER** erklärt im Video-Kurs „Lightroom perfekt im Griff“ alles, was Sie über den Bildverwalter wissen müssen: Bildfehler korrigieren, per Stapelverarbeitung effektiv arbeiten und HDR-Serien erstellen.

WEITERE KURSE ZUM THEMA FOTO



Photoshop Elements im Überblick

SVEN FISCHER 175 MINUTEN 19,90 €

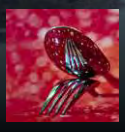


Foto-Projekte: Schritt für Schritt zum perfekten Bild

CHIP FOTO-VIDEO 215 MINUTEN 19,90 €

Photoshop für Fotografen

HEICO NEUMEYER 182 MINUTEN 19,90 €

Entfesselt blitzen: So geht‘s richtig

HAUKE SEYFARTH 155 MINUTEN 19,90 €

Alle Kurse sind als Online-Stream jederzeit verfügbar - auf Handy, Tablet & PC

www.chip.academy/foto

DIE NÄCHSTE AUSGABE VON MagPi ERSCHEINT AM 10. MAI 2017

PROGRAMMIEREN LERNEN MIT DEM PI

Thorsten Franke-Haverkamp, Redaktionsleiter MagPi

Eine der wunderbaren Sachen beim Raspberry Pi ist, dass er es einem wirklich leicht macht, Dinge einfach auszuprobieren. Wenn Sie etwa schon immer mal einen Server administrieren wollten: Legen Sie los! Oder wenn Sie eine LED-Schaltung selbst entwerfen möchten: Nur zu! Sollte etwas schiefgehen oder Sie sich total verkonfigurieren, dann starten Sie einfach neu. So einfach ist das.

Das Gleiche gilt fürs Programmieren. Kaum eine andere Plattform macht es Ihnen so leicht, Ihre ersten Schritte beim Coden zu wagen. Dabei können Sie mit dem Pi Ihren Programmcode auf simple, aber dennoch beeindruckend effiziente Weise mit Hardware verknüpfen. In unserem großen Schwerpunkt ab Seite 16 führen wir Sie in die Grundlagen des Programmierens ein. Vorkenntnisse sind nicht erforderlich! Dabei konzentrieren wir uns vor allem auf Python, weil diese Sprache für Einsteiger ideal ist. Ihre Syntax ist leicht zu lesen und umzusetzen. Dennoch lassen sich mit Python die tollsten Projekte auf dem Raspberry Pi verwirklichen. Probieren Sie es aus – der Einstieg ist wirklich ganz einfach.

Viel Spaß mit dieser Ausgabe!

HIGHLIGHT AUF DVD: DEBIAN MIT PIXEL ZUM AUSPROBIEREN

RasPi-Feeling auf dem PC Sie wollen Debian mit Pixel gefahrlos ausprobieren? Dann starten Sie unsere Live-DVD

Genial: Das (nahezu) gleiche Betriebssystem wie auf dem Raspberry Pi können Sie nun auch auf Ihrem Desktop-Rechner oder Notebook erleben. Unsere Heft-DVD sollten Sie daher unbedingt einmal ausprobieren – auf ihr findet sich nämlich Debian mit Pixel-Desktop als Live-System. Sie müssen also lediglich die DVD einlegen und von dieser booten. Wie das geht und wie Sie das System dauerhaft auf einem USB-Stick einrichten, beschreiben wir ab **Seite 66**.

PI ZERO FÜR 1 EURO ZU JEDEM ABO! SEITE 14

MAGPI KOSTENLOS

Sie haben die ersten Ausgaben 2016 verpasst? Kein Problem, ab sofort können Sie die Ausgaben 5/2016 und 6/2016 kostenlos im CHIP Kiosk als PDF herunterladen (einmalige kostenlose Registrierung erforderlich). **chip-kiosk.de/chip/magpi**

MAGPI IM ABONNEMENT

Gefällt Ihnen die aktuelle Ausgabe von MagPi, dem offiziellen Raspberry-Pi-Magazin? Das Heft gibt es auch im Abonnement. So verpassen Sie keine Ausgabe mehr und bekommen das Magazin alle zwei Monate bequem frei Haus geliefert. Damit sparen Sie nicht nur Geld, sondern sichern sich auch eine tolle Abo-prämie: Zu jedem Abo gibt es den Raspberry Pi Zero mit attraktivem Kabel-Bundle. Mehr dazu auf **S. 14**

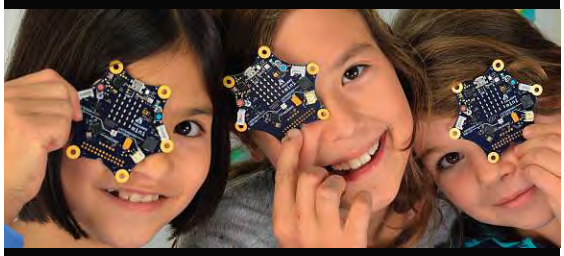
Service

Inhalt

Ausgabe 2 März/April 2017 magpi.de

TRENDS

Calliope Mini 07



Der Mikrocontroller aus Berlin soll die digitale Bildung an deutschen Schulen vorantreiben

> WATERSCOPE 06
Ein Mikroskop aus dem 3D-Drucker

> PI ZERO / CALLIOPE MINI 07
Endlich erhältlich: Pi Zero & ein Mikrocontroller aus Berlin

> WD PI DRIVE / STAR TREK PI 08
Star-Trek-Star Wil Wheaton und sein Homeserver

> HANS-SAUER-PREIS 09
Nominierung für Düsseldorfer Codingschule

> THE SOUNDS OF NEW YORK 10
Dem Lärm der Großstadt auf der Spur

> COMPUTE MODULE 3 12
Das neue Modul auf Basis des Pi 3 ist 10 Mal schneller

Schwerpunkt Programmieren

> EINSTEIGER-GUIDE CODING 16
Legen Sie einfach los mit dem Programmieren

> WAS IST EIN PROGRAMM? 18
Die Grundlagen: So ist Software aufgebaut und das passiert im Hintergrund

> VARIABLEN 20
So speichern Sie Daten und Objekte

> SCHLEIFEN MIT WHILE & FOR 22
Damit macht Ihr Programm die Arbeit fast von allein

> BEDINGTE ANWEISUNGEN 24
Machen Sie Ihren Code intelligenter und selbstständiger

> SO TICKEN FUNKTIONEN 26
Erstellen Sie blockweise Code für stabilere Programme

> CODE IMPORTIEREN 28
So verwenden Sie vorhandenen Code als Basis

PROJEKTE

> SISYPHUS 30
Dieser Tisch ist ein echtes Kunstwerk

> RASPI-GESTEUERTE TÜR 34
Gesichtserkennung mit Windows IoT

> PEGASUS 36
Temporekorde mit einem umgebauten Düsenjet

> QBEE 40
Ein Kleid wird zur innovativen Social-Media-Plattform

> SELBSTSPIELENDE ORGEL 42
Diese Orgel spielt jede beliebige MIDI-Datei ab

> WASSERSTANDSANZEIGE 44
Immer im Blick: Der Füllstand der Regentonne

> RASPCADE: DAS GEHÄUSE 46
Im 4. Teil der Serie setzen wir das Gehäuse zusammen

> BODENKAMERA 48
Eine moderne Version eines NASA-Klassikers

> GERO – DER „GERMAN ROBOT" 50
Ein Pi-betriebener Robotor made in Germany

PROGRAMMIEREN 16

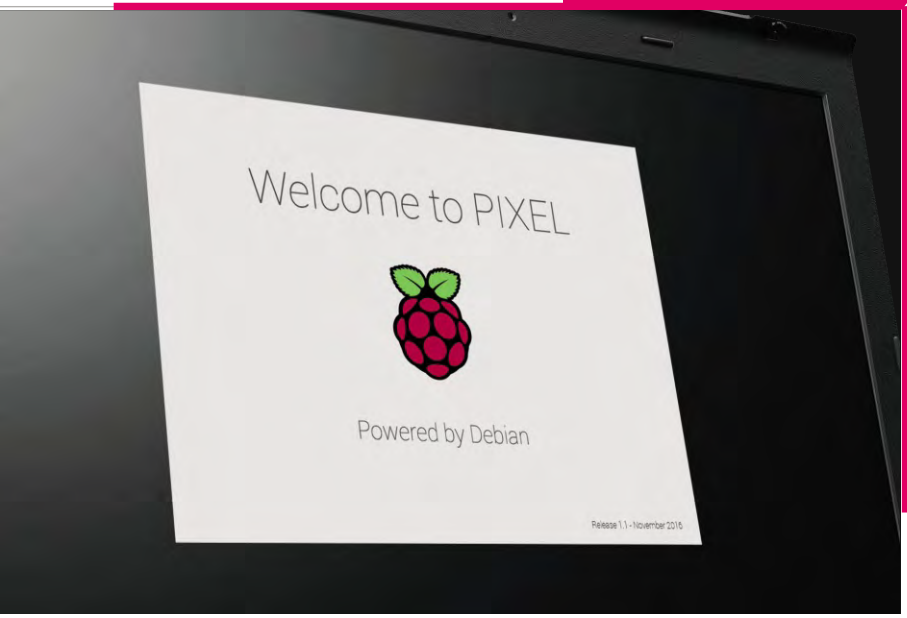
Entdecken Sie für sich die Welt des Programmierens – unser Einsteigerkurs macht es leicht

Sisyphus

Eine Kugel malt fantastische Figuren in den Sand – gesteuert durch einen starken Magneten

30

HEFT-DVD 66



DEBIAN & PIXEL

Unsere Heft-DVD ist diesmal bootfähig!
Denn Debian & Pixel ist ein Live-Betriebssystem, das Sie auf einem normalen PC verwenden können. Stecken Sie einfach die Heft-DVD in Ihr Laufwerk, ändern Sie die Bootreihenfolge und starten Sie dann den Rechner.

> PIXEL AUF DEM DESKTOP 66
> PC-START MIT DER DVD 68
> ERSTE SCHRITTE MIT DEBIAN 70
> USB-STICK MIT PIXEL 72

PRAXIS

> DAS STECKBRETT 52
So nützlich und vielseitig ist ein Breadboard

> GPIO FÜR EINSTEIGER 54
Verständlich erklärt: So nutzen Sie die Pins

> FERNSTEUERUNG PER VNC 58
So steuern Sie Ihren Pi von anderen Rechnern aus

> PROGRAMMIEREN IN C 60
Teil 4 der Serie: Flusskontrolle

> EIN BESONDERER WECKER 62
Lassen Sie sich mit den ersten Sonnenstrahlen wecken

> PARALLEL PROGRAMMIEREN 64
So nutzen Sie die gesamte Rechenleistung Ihres Pi

> MUSIKBOX AUF KNOPFDRUCK 74
Spielen Sie mit zwei oder mehr Knöpfen Sounds ab

> BOINC HILFT DER FORSCHUNG 76
Spenden Sie Rechenkapazität Ihres Pi

> GOOGLE DEEPDREAM 78
Surreale Traumbilder auf dem Pi

> PISPY-CAM 80
Eine Überwachungskamera mit Bewegungssensor

> SPIELEN MIT SCRATCH 82
Programmieren Sie ein kleines Reaktionsspiel

> EMPATHYBOT 84
Dieser Pi-Roboter erkennt Ihre Gefühle

> RENNFIEBER AUF DEM PI 86
Ein Pferderennen mit außergewöhnlicher Steuerung

> FAQ: GPIO-PINS 92
Die wichtigsten Fragen und Antworten

ZUBEHÖR

> MICROBOT V2 94
> PROTOBOARD 96
> MOTE PHAT 97
> LIBREELEC 7.0 98
> SERVO KIT 100
> PAN-TILT HAT 101
> RASPIROBOT ROVER KIT 102
> NEXTCLOUD-BOX 104

ZUM SCHLUSS

> BUCHEMPFEHLUNGEN 106

> COMMUNITY-PROFIL 108
Jillian Ogle ist die Herrin über abenteuerliche Roboter

> ZIELE 2017/CROWDFUNDING 110

> VERANSTALTUNGSKALENDER 112
Die wichtigsten Termine im deutschsprachigen Raum

> LESERBRIEFE 113

> KOMMENTAR 114
Das Leben ist ein permanenter Lernprozess

SERVICE

> EDITORIAL 3
> HEFT-DVD 67
> IMPRESSUM 93

WATERSCOPE QUALITÄTSPRÜFUNG MIT DEM PI

Ein Start-up der Universität von Cambridge sagt dem Trinkwassermangel den Kampf an – mit einem Mikroskop aus dem 3D-Drucker

Wasser, vor allem trinkbares Wasser, ist lebensnotwendig. Für uns ist es selbstverständlich, den Wasserhahn aufzudrehen, wenn wir etwas trinken möchten oder damit Geschirr spülen oder das Fahrrad putzen. Dazu verwenden wir frisches Trinkwasser. Das ist ein Luxus, der für viele Millionen Menschen unvorstellbar ist. Nach Schätzungen der Weltgesundheitsorganisation (Stand November 2016) trinken weltweit fast zwei Milliarden Menschen kontaminiertes Wasser.

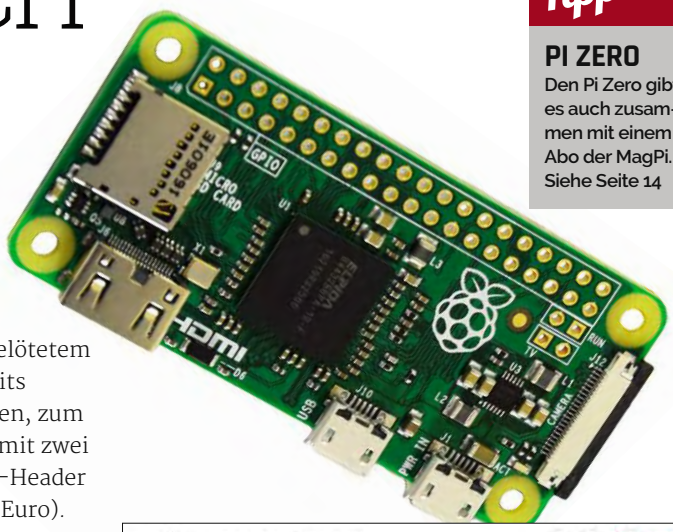
Gerade in den Entwicklungsländern ist die schnelle Prüfung der Wasserqualität eine Herausforderung. An dieser Stelle will WaterScope helfen (**waterscope.org**). Die Firma ist eine Ausgründung der Universität von Cambridge. Das Start-up hat ein besonders günstiges System zum Nachweis von Krankheitserregern wie etwa E.coli, Listerien und Legionellen konzipiert. Damit sollen verlässliche Resultate bereits nach 1–2 Stunden vorliegen – 20-mal schneller als mit herkömmlichen Methoden.

Vor Ort einsetzbar

Die Prüfung besteht aus drei Teilen. Teil eins ist ein spezielles Test-Kit für Bakterien, mit dem die Wasserproben vor Ort filtriert und untersucht werden können. Nachdem die Bakterien in den Samples Zeit hatten heranzureifen, werden mit einem Open-Flexure-Mikroskop Fotos von dem Sample gemacht. Die Teile des Mikroskops lassen sich mit einem 3D-Drucker ausdrucken. Es wird an einen Raspberry Pi mit einem V2-Kameramodul angeschlossen. Nach der Aufnahme werden die Bilder mittels spezieller Bildbearbeitung optimiert und mit einem Archiv abgeglichen. Das Mikroskop ist im Shop erhältlich, aber als Open-Source-Projekt unter der CERN-Open-Hardware-Lizenz freigegeben. Die STL-Files für den 3D-Druck befinden sich auf GitHub (**bit.ly/2lo2DC5**). Eine Schritt-für-Schritt-Bauanleitung auf Englisch finden Sie hier: **bit.ly/2kCgw8A**.

PI ZERO ENDLICH ERHÄLTLICH

Es ist so weit – der Mini unter den Raspberry Pis ist nun endlich in Deutschland zu kaufen



Tipp

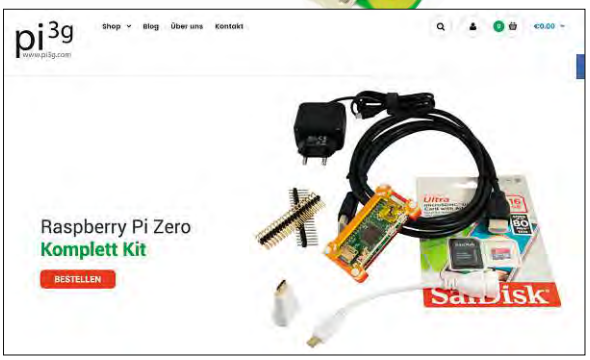
PI ZERO

Den Pi Zero gibt es auch zusammen mit einem Abo der MagPi. Siehe Seite 14

Bisher mussten deutsche Maker den Pi Zero aus England selber einführen, war der Winzling doch nur in UK und den USA erhältlich. Wer außer dem Pi Zero und keine weitere Hardware benötigte, überlegte sich zweimal, ob sich das lohnt. Dazu kommt, dass der Versand aus England teilweise recht lange dauert.

Damit ist endlich Schluss. Der Zero ist nun auch in Deutschland erhältlich, und zwar exklusiv über das Portal **buyzero.de**. Aufgrund der hohen Nachfrage ist die Abgabe auf eine Platine pro Haushalt beschränkt. Neben dem eigentlichen Pi Zero mit oder ohne eingelötetem Pin-Header werden auch Kits inklusive Zubehör angeboten, zum Beispiel Pi Zero Essentials mit zwei USB-Adaptern und 40-Pin-Header zum selbst einlöten (ca. 14 Euro). Daneben gibt es weiteres Zubehör, etwa verschiedene Hats.

Der Anbieter, Pi3g (**pi3g.com**), ist ein Start-up, das sich ursprünglich auf Komplett-Kits mit dem Raspberry Pi spezialisert hat, zum Beispiel mit Kamera oder LCD-Touchdisplay plus passendem Case und vorkonfigurierter Software. Auch eine Box, die das Internetsurfen anonymisiert, ist dabei.



EIN CALLIOPE MINI FÜR JEDEN DRITTKLÄSSLER

Der Mikrocontroller aus Berlin steht für digitale Souveränität und Spaß am Coden und soll Starthilfe für die digitale Bildung werden

Das Team hinter dem Calliope Mini (**calliope.cc**) hat ein ambitioniertes Ziel: Alle Schüler der dritten Klasse sollen jedes Jahr flächendeckend einen Mikrokontroller bekommen.

„Die digitale Bildung steckt in Deutschland noch in den Kinderschuhen – unsere Nachbarländer der Europäischen Union sind in vielen Fällen schon weiter“, erklärt Gesche Jost, Professorin für Designforschung an der Universität der Künste in Berlin. Sie ist einer der Köpfe hinter dem Projekt. Um das Ziel zu erreichen, arbeitet das Team unter anderem mit Ausbildungseinrichtungen und Bildungsministerien in den Ländern zusammen. Die kleine Platine, die mit dem BBC micro:bit kompatibel ist, ist extrem flexibel. Sie ist mit 25 roten und einer RGB-LED, zwei programmierbaren Knöpfen, einem kombinierten Lage- und Bewegungssensor, einem Kompass und einem Bluetooth-Modul ausgestattet. Alle Ecken haben spezielle Fähigkeiten. Zwei Ecken sind etwa mit je einem Plus und einem Minus gekennzeichnet. Mithilfe von Klemmen entsteht so eine Batterie. Code kann via PC aber auch kabellos per App übertragen werden.

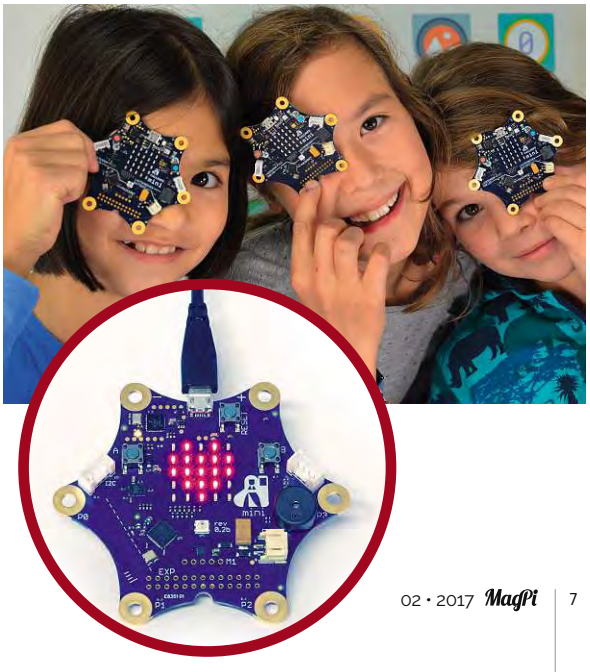


Foto: Jørn Alraun (Calliope); Pi3g

STAR TREK SERVER

Schauspieler und Star-Trek-Fan im Pi-Fieber

Der Raspberry Pi konnte einen weiteren prominenten Fan gewinnen: Wil Wheaton. Wil ist vor allem Star-Trek-Fans als Wesley Crusher aus Star Trek – die nächste Generation bekannt. Der Schauspieler und Betreiber eines der größten Star-Trek-Fanblogs hat sich einen Raspberry Pi Home Mediaserver gebaut und ist begeistert. Unter **@wilw** hat er von seinem Projekt auf Twitter berichtet: „Ich habe gerade jede Menge Spaß mit meinem Raspberry-Pi-Wochenendprojekt." Er orientierte sich bei seinem Projekt an einem Online- Tutorial von Entwickler Mel Grubb (**tinyurl.com/PiHomeServer**).

Wil verriet auch sein Setup. Er verwendete einen Raspberry Pi 3 mit Raspbian und eine externe Festplatte. Auf die Frage eines seiner Follower, ob man damit eventuell auch die Effizienz eines Warp Core erhöhen könnte, witzelte Wil zurück: „Klar, Du musst nur das richtige Repository zu apt.sources hinzufügen."

Wil Wheaton hofft, dass sein neuer Home-Server lange und in Frieden leben wird

© Luigi Novi / Wikimedia Commons

Rechts Die neuen PiDrive-Geräte werden mit microSD-Karte mit Noobs ausgeliefert

WD PIDRIVE FOUNDATION EDITION

Mit Noobs als Bonus auf SD-Karte

Die dritte Generation der für den Raspberry Pi optimierten PiDrive-Produkte von WDLabs ist seit Kurzem verfügbar. Die Foundation Edition gibt es wahlweise als 64 Gbyte USB-Flash-Speicher mit USB 2.0 für ca. 20 Euro oder als USB-Festplatte inklusive WD PiDrive-Kabel mit 250 Gbyte oder 375 Gbyte Speicher ab 31,90 Euro. Die Edition ähnelt damit den Vorgängermodellen. Die Geräte werden alle mit einer microSD-Karte mit Noobs ausgeliefert.

Die WD PiDrive-Geräte sind dafür gedacht, das Betriebssystem auf dem USB-Flash-Speicher oder auf der Festplatte laufen zu lassen. So haben Sie genug

> „Die dritte Runde für die praktischen PiDrive-Speicher"

Platz, um mehrere Raspberry-Pi-Projekte gleichzeitig zu hosten. Ebenfalls weiterhin erhältlich ist die Berry-Boot Edition mit 1 TByte Speicher und einer microSD-Karte, auf der BerryBoot vorinstalliert ist. Mit Berry-Boot können Sie mehrere Linux-Distributionen parallel auf einem USB-Speicher oder einer SD-Karte speichern und im Bootmenü auswählen.

CODINGSCHULE FÜR DEN HANS-SAUER-PREIS NOMINIERT

Ein engagiertes Coding-Lab aus Düsseldorf ist im Finale des Wettbewerbs, der Projekte zur Schulentwicklung auszeichnet

Was benötigt eine Schule, um digitale Bildung zu vermitteln? Nicht mehr als einen Raum, ein Konzept, ein paar Raspberry Pis, etwas Open-Source-Software und -Lehrmaterialien sowie vor allem Mut, etwas Neues zu wagen. Davon ist Tobias Hübner, Lehrer am Luisen-Gymnasium in Düsseldorf, überzeugt. Seit vielen Jahren verfolgt er das Ziel, digitale Kompetenzen zu vermitteln.

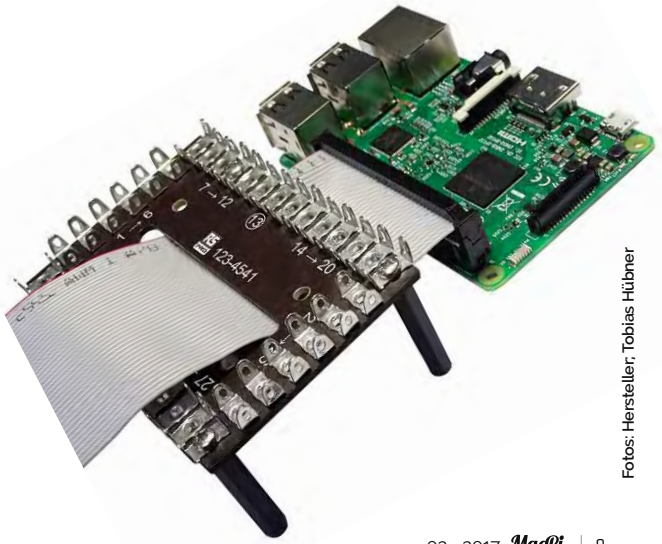
Sein aktuelles Projekt: Eine Codingschule nach dem Vorbild von Maker-Labs, die mit allem ausgestattet ist, was man zum Programmieren und dem Umgang mit Elektronik benötigt. Hier sollen etwa Kinder mithilfe des Raspberry Pi lernen, wie Computer funktionieren und wie man sie programmiert. Außerdem können Schülerinnen und Schüler im Rahmen von Projekttagen mit den Kompetenzen des für alle Schulen in NRW verbindlichen „Medienpasses NRW“ vertraut gemacht werden.

Das Projekt „Codingschule“ ist bereits recht konkret, geeignete Räume wurden gefunden, Kooperationspartner an Bord geholt. Das Konzept wurde überdies bei der Hans-Sauer-Stiftung eingereicht, die jährlich einen mit 5.000 Euro dotierten Preis für Projekte im Rahmen der Schulentwicklung auslobt. Das Finale des Wettbewerbs findet vom 9. bis 12.3. in München statt.

NEUE LÖTPLATINE FÜR DEN RASPBERRY PI

Das Solder-Board des Elektronikhändlers RS Components erlaubt die kostengünstige Entwicklung von Prototypen für den Pi

Die RS-Pro-Lötplatine besitzt insgesamt 40 Lötkontakte, die genau zu den 40 GPIO-Pins des Raspberry Pi passen. Dadurch wird das Löten und Entlöten von elektronischen Komponenten ermöglicht, ohne direkt auf der Raspberry-Pi-Platine zu arbeiten. Geht beim Löten etwas schief, lässt sich die Lötplatine einfach ersetzen – bei einem Stückpreis von 2,59 Euro ist das kein großer finanzieller Verlust. Die Bohrungen in der RS-Pro-Platine sind auf die Montagelöcher des Raspberry Pi ausgerichtet. Somit kann die Lötplatine direkt unter dem Pi angebracht werden. Dazu ist allerdings ein zusätzlicher Montagesatz nötig, der aus vier Beinen und acht Schrauben besteht. Dieser ist nicht im Lieferumfang enthalten. Die zentrale Öffnung im Solder-Board erlaubt die Durchführung eines Bandkabels (ebenfalls nicht im Lieferumfang). Weitere Informationen erhalten Sie unter **http://bit.ly/2lLHXwZ**.



Fotos: Hersteller, Tobias Hübner

THE SOUNDS OF NEW YORK CITY

Die New Yorker Uni geht mit Hilfe des Pi dem Lärm im Big Apple auf den Grund

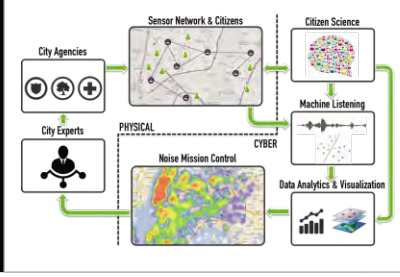
Rechts New York ist eine der lebendigsten Städte der Erde. Doch Lärmbelästigung ist ein großes Problem für die Bewohner

Sounds of New York City ist ein Projekt der New York University (NYU). Ziel ist, die Lärmbelästigung in der Stadt zu beobachten und zu dokumentieren. Eine Reduzierung der Lärmbelästigung ist, laut NYU, einer der wichtigsten Faktoren, wenn es um die Verbesserung der Lebensqualität der Stadtbewohner in den USA geht. Schätzungen zufolge sind neun von zehn Erwachsene in New York City (NYC) zu hohen Lärmpegeln ausgesetzt.

Um dieses Problem genauer zu analysieren,hat die NYU Raspberry Pis mit Mikrofonen bestückt und in New York verteilt, um den Lärmpegel zu messen. Charlie Mydlarz, ein leitender Wissenschaftler des Projekts: „Wir arbeiten eng mit den Behörden zusammen, unter anderem mit der Abteilung für Umweltschutz. Mit den von uns gesammelten Daten wird es möglich, die geltenden Regeln zum Lärmschutz effizienter durchzusetzen. Außerdem arbeiten wir mit dem Gesundheitsministerium zusammen, um die Beziehung zwischen der Gesundheit der Bürger und dem herrschenden Lärmpegel der Stadt zu untersuchen.“

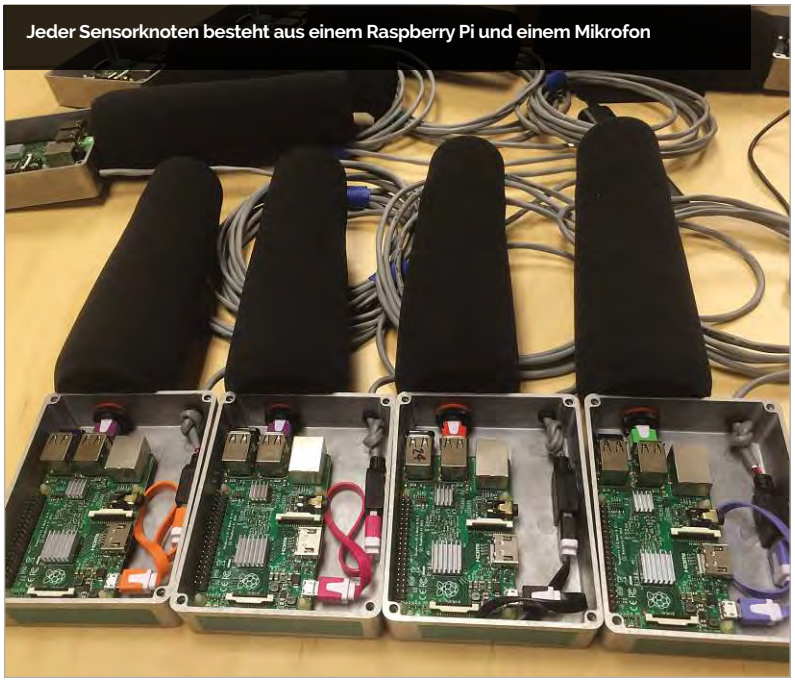
LÄRMQUELLEN IDENTIFIZIEREN

Die von den Raspberry Pis gesammelten Daten werden dazu genutzt, eine Kontrollstation einzurichten. Die Daten aus dem Netzwerk fließen direkt in die Infrastruktur, die die Daten auswertet. Maschinelle Lernmethoden werden verwendet, um Verursacher des Lärms, wie etwa Polizeisirenen, zu identifizieren.



Pi 2 als Sensorknoten

Mydlarz erklärt: „Jeder Sensorknoten besteht aus einem Raspberry Pi 2B, auf dem ein Debian Jessie ohne grafische Oberfläche



Jeder Sensorknoten besteht aus einem Raspberry Pi und einem Mikrofon

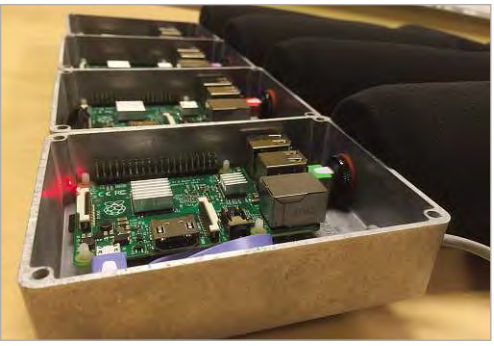
> Schätzungen zufolge sind neun von zehn Erwachsene in New York City zu hohen Lärmpegeln ausgesetzt

läuft. Die Sensorknoten haben keinen offiziellen Namen, aber ich nenne sie Acoustic Sensing Devices oder kurz ASDs."

An den Raspberry Pi ist ein digitales Mikrofon angeschlossen. Alle Daten werden mittels WiFi an das Netzwerk der NYU übertragen. Dabei hat das Team sehr genau darauf geachtet, dass die Privatsphäre der Bürger nicht beeinträchtigt wird. Die Audiodaten werden als 10 Sekunden lange, zufällig ausgewählte Ausschnitte gesammelt. Die Auschnitte werden mit dem verlustfreien FLAC-Encoder komprimiert und dann mit AES- und RSA-Verschlüsselung versehen.

„Wir haben viel Arbeit darauf verwendet, die Privatsphäre zu schützen. Sogar eine externe Beraterfirma haben wir beauftragt, um zu überprüfen, dass normale Unterhaltungen auf der Straße nicht von den Mikrofonen aufgenommen werden können", so Mydlarz. „Man müsste schon einen der Sensorknoten direkt anschreien, damit die entsprechenden Sätze verständlich aufgenommen würden – das wäre dann keine private Unterhaltung mehr."

Jeder Sensorknoten wurde zudem mit einem Hinweis gekennzeichnet, um Passanten über das Projekt zu informieren. Im Sommer 2016 waren bereits 20 dieser Knoten in Betrieb, bis zum Frühjahr 2017 sollen es 100 Stück werden. Die Sensoren werden extreme Wetterunterschiede verkraften müssen: sehr feuchte, heiße Sommer, aber auch klirrende Kälte im Winter.



Oben Die New York University plant, bis zum Frühjahr 2017 insgesamt 100 Geräte in der Innenstadt aufzustellen

RASPBERRY PI 2016

2016 war ein extrem erfolgreiches Jahr für den Raspberry Pi und die Foundation. Neben dem großen Erfolg des Boards wurden auch zahlreiche neue Initiativen ins Leben gerufen.

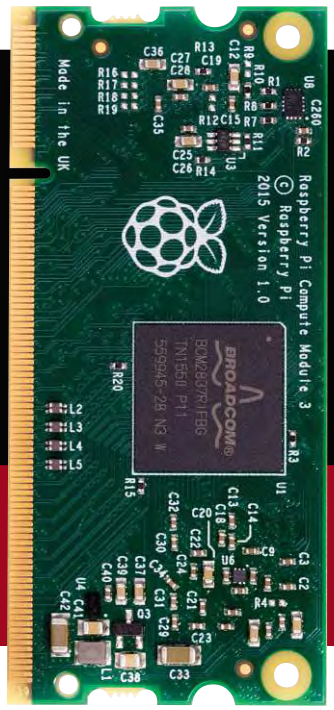
11 MILLIONEN
VERKAUFTE RASPBERRY PIS

RASPI 3
SEIT FEBRUAR 2016

DEBIAN + PIXEL
AUCH FÜR PC & MAC

MAGPI
ERSCHEINT ERSTMALS IN DEUTSCH

COMPUTE MODULE 3 VERFÜGBAR



Neues Modell auf Basis des Pi 3 ist zehn Mal leistungsfähiger

Die Raspberry Pi Foundation hat die neue Version des Compute Module auf den Markt gebracht. Das Compute Module ist ein Raspberry Pi im Miniaturformat für Entwickler und passt in einen Standard-DDR2-SODIMM-Sockel. Das neue Compute Module 3 (CM3) basiert auf dem Pi 3 und bringt deshalb gegenüber seinen Vorgängern einen großen Leistungsschub.

„Es hat den gleichen BCM2837-Prozessor, der bis zu 1,2 GHz hochtaktet, und 1 Gbyte RAM“, sagt der COO der Foundation, James Adams. Das bedeutet, verglichen mit dem ersten Compute Module (CM1) bietet das neue CM3 den doppelten Arbeitsspeicher und etwa zehn Mal soviel Leistung. Wie schon der Pi 3 unterstützt auch das CM3 Windows 10 IoT Core, was für Entwickler ganz neue Möglichkeiten eröffnet:

> „CM 3 gibt es in zwei Versionen – mit und ohne integrierten Flash-Speicher“

„Ein IoT-Projekt kann einfach von einem Pi 3 auf ein benutzerdefiniertes, CM3-basiertes System übertragen werden.“ Eine Schwachstelle des CM1 war bisher der fest verbaute eMMC-Flash-Speicher. „Einige Nutzer wollten vollen Zugriff, um den Speicher selbst erweitern zu können“, erklärt James. Deshalb gibt es diesmal zwei Versionen des CM3: eine mit 4 GByte eMMC-Speicher und eine Lite-Version, die die Nutzer selbst um einen SD-Kartensteckplatz oder eMMC-Flash-Speicher erweitern müssen. „Wir rechnen damit, dass die Lite-Version sehr beliebt sein wird.“

Das Compute Module 3 kommt etwa in der neuen Modellreihe von großformatigen „smarten“ Displays von NEC zum Einsatz

Schnell zum Upgrade

Mit ein paar Einschränkungen kann das CM3 das CM1 1:1 ersetzen, da beide Pin-kompatibel sind. Das CM3 ist jedoch 1 mm höher. Unter Volllast hat es zudem einen höheren Stromverbrauch und wird auch heißer. In Deutschland bietet RS Components die Lite-Version für 26, die Speicher-Version für 32 Euro an.

Parallel gibt es ein aktualisiertes IO-Breakout-Board (CMIO3, ca. 115 Euro), das das CM1, das CM3 und das CM3 Lite unterstützt.

KOSOVO PI WARS

Jugendliche im Kosovo lassen ihre Roboter gegeneinander antreten

Junge Maker aus dem Kosovo veranstalteten im Dezember 2016 die ersten Kosovo Pi Wars. Der Kosovo ist immer noch dabei, den Krieg von 1998/1999 aufzuarbeiten. „Vor acht Jahren hat sich Kosovo von Serbien losgesagt und seine Unabhängigkeit erklärt“, sagt Andy Moxon, ein britischer Ingenieur, der seit zweieinhalb Jahren im Kosovo lebt. Andy leitet in der Kleinstadt Shtime im Süden Kosovos einen kleinen Coding-Club, die Young Innovators (**magpi.cc/2iHAtuc**). Obwohl die Zeiten immer noch schwierig sind, finden sich auch hier Computer, Roboter und jede Menge Raspberry Pis. „Der Coding-Club ist dafür gedacht, den Schülern Mathematik und Physik spielerisch nahezubringen und sie gleichzeitig in die Grundlagen des Programmierens und der Robotik einzuführen“, erklärt Andy. „Um dieses Ziel zu erreichen, arbeiten wir mit ganz einfachen Robotern.“

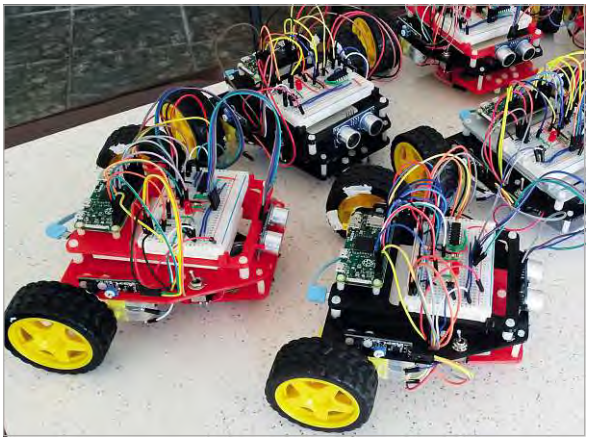
Jeder Roboter besteht aus einem Pi Zero, der von einer Powerbank versorgt wird und zwei Motoren, die von einem L293D Motortreiber gesteuert werden, dazu Ultraschall-Abstandssensoren sowie Infrarot-Zeilensensoren.

„Wir verwenden zwei zusätzliche Infrarotsensoren, um die Rad-Umdrehungen zu messen“, so Andy weiter, „dazu haben wir mit Nagellack weiße Streifen auf die Räder gemalt.“ Dank dieser Sensoren sind die Roboter für sehr interessante autonome Herausforderungen geeignet.

Für große Begeisterung sorgte die Anschaffung eines Ultimaker 2+ 3D-Druckers, mit dem die Schüler die mit FreeCAD konstruierten Chassis für die Roboter ausdrucken und so nebenbei etwas über 3D-Prototyping lernen konnten. Die Roboter durften sich im Dezember in den ersten Kosovo Pi Wars messen. Das Event hat es sogar ins Lokalfernsehen geschafft. „Ein großes Dankeschön an die Raspberry Pi Community, ohne die wir das niemals hätten auf die Beine stellen können“, bedankt sich Andy, der vor einem Jahr das erste Mal mit einen Raspberry Pi gearbeitet hat.

Oben Young Innovators ist ein Schülerclub für Robotik in Shtime, Kosovo

Unten Die Roboter der Young Innovators, die bei den Pi Wars angetreten sind



6 x MagPi +
Pi Zero-Bundle sichern!

Ihre Vorteile

✗ Mehr Komfort
Pünktliche, bequeme und kostenlose Lieferung
Eine spannende DVD in jedem Heft

✗ Ein Heft gratis
Bezahlen Sie bequem per Bankeinzug und Sie erhalten zusätzlich eine Ausgabe MagPi gratis!

✗ Attraktives Dankeschön
Freuen Sie sich auf ein hochwertiges Produkt als Dankeschön!

MagPi

Raspberry Pi Zero

• 1 Ghz, Single-core CPU Prozessor
• 512MB RAM • Mini-HDMI für 1080p60-Video-Output • 2x Micro-USB-Anschluss (1x für Daten, 1x für Strom) • 40-Pin-GPIO
• Inklusive USB-Konverter-Kabel, HDMI-Konverter und Kamera-Kabel.

Exklusiv nur bei uns

Ausfüllen und abschicken

oder unter

services.chip.de/abo/pi-maerz

bestellen

So einfach können Sie bestellen:
(Telefon) 0781-639 45 26
(Fax) 0781-846 19 1
(E-Mail) abo@chip.de
(URL) services.chip.de/abo/pi-maerz

Weitere Angebote finden Sie unter **www.chip-kiosk.de/chip**

Sie haben ein gesetzliches Widerrufsrecht, die Belehrung können Sie unter www.chip-kiosk.de/widerrufsrecht abrufen.

CHIP erscheint im Verlag: CHIP Communications GmbH, St.-Martin-Straße 66, 81541 München.
Geschäftsführung: Thomas Koelzer (CEO), Markus Scheuermann (COO) Handelsregister: AG München, HRB 136615. Die Betreuung der Abonnenten erfolgt durch: Abonnenten Service Center GmbH, CHIP Aboservice, Hubert-Burda-Platz 2, 77652 Offenburg. Der Verlag behält sich vor, Bestellungen ohne Angabe von Gründen abzulehnen.

◯ Ja, ich bestelle 6 x MagPi für nur 54,80 € (inkl. MwSt. und Porto). **M17MA02P2**

Zunächst für ein Jahr (6 Ausgaben). Das Dankeschön erhalte ich umgehend nach Zahlungseingang. Das Abo kann ich nach Ablauf eines Jahres jederzeit wieder in Textform kündigen. Es genügt eine kurze Nachricht von mir an den CHIP Aboservice, Postfach 225, 77649 Offenburg oder per E-Mail an abo@chip.de. Dieses Angebot gilt nur in Deutschland (Konditionen für das Ausland bitte auf Anfrage unter abo@chip.de) und nur solange der Vorrat reicht.
Für Zahlungen per SEPA-Lastschrift aus dem Ausland oder bei Bestellungen ins Ausland hilft Ihnen unser Aboservice unter 0781/6394526 oder per Mail an abo@chip.de gerne weiter.

Name, Vorname

Straße, Haus-Nr.

PLZ,Ort

Telefon/Handy Geburtsdatum

E-Mail

und erhalte als Dankeschön dazu

Bundle Raspberry Pi Zero + HDMI-Konverter sowie USB- und Kamera-Kabel (C997), Zzgl. 1 € Zuzahlung

Ich bezahle bequem durch Bankeinzug, erhalte eine Ausgabe gratis vorab und mein Geschenk sofort SEPA-Lastschriftmandat: Ich ermächtige die CHIP Communications GmbH, wiederkehrende Zahlungen von meinem Konto mittels Lastschrift einzuziehen. Zugleich weise ich mein Kreditinstitut an, die vom Verlag auf mein Konto gezogenen Lastschriften einzulösen. Hinweis: Ich kann innerhalb von acht Wochen, beginnend mit dem Belastungsdatum, die Erstattung des belasteten Betrags verlangen. Es gelten dabei die mit meinem Kreditinstitut vereinbarten Bedingungen

D E
IBAN Ihre BLZ Ihre Konto-Nr.

Zahlungsempfänger:
CHIP Communications GmbH, St.-Martin-Straße 66, 81541 München
Gläubiger-ID: DE11ZZZ00000186884
Mandatsreferenz wird separat mitgeteilt.

Mit folgender Kreditkarte: ◯ VISA ◯ Eurocard/Mastercard

Kreditkarten-Nr. Prüfnr.

Gültig bis:

◯ Ja, ich bin einverstanden, dass die CHIP Communications GmbH mich per E-Mail über interessante Vorteilsangebote informiert. Meine Daten werden nicht an Dritte weitergegeben. Dieses Einverständnis kann ich selbstverständlich jederzeit widerrufen.

Datum Unterschrift

Coupon ausschneiden und schicken an: **CHIP Aboservice, Postfach 225, 77649 Offenburg**
oder im Internet bestellen unter: services.chip.de/abo/pi-maerz **M17MA02P2**

Programmieren lernen

MIT DEM PI

Entdecken Sie die Kunst des Programmierens mit dem Raspberry Pi

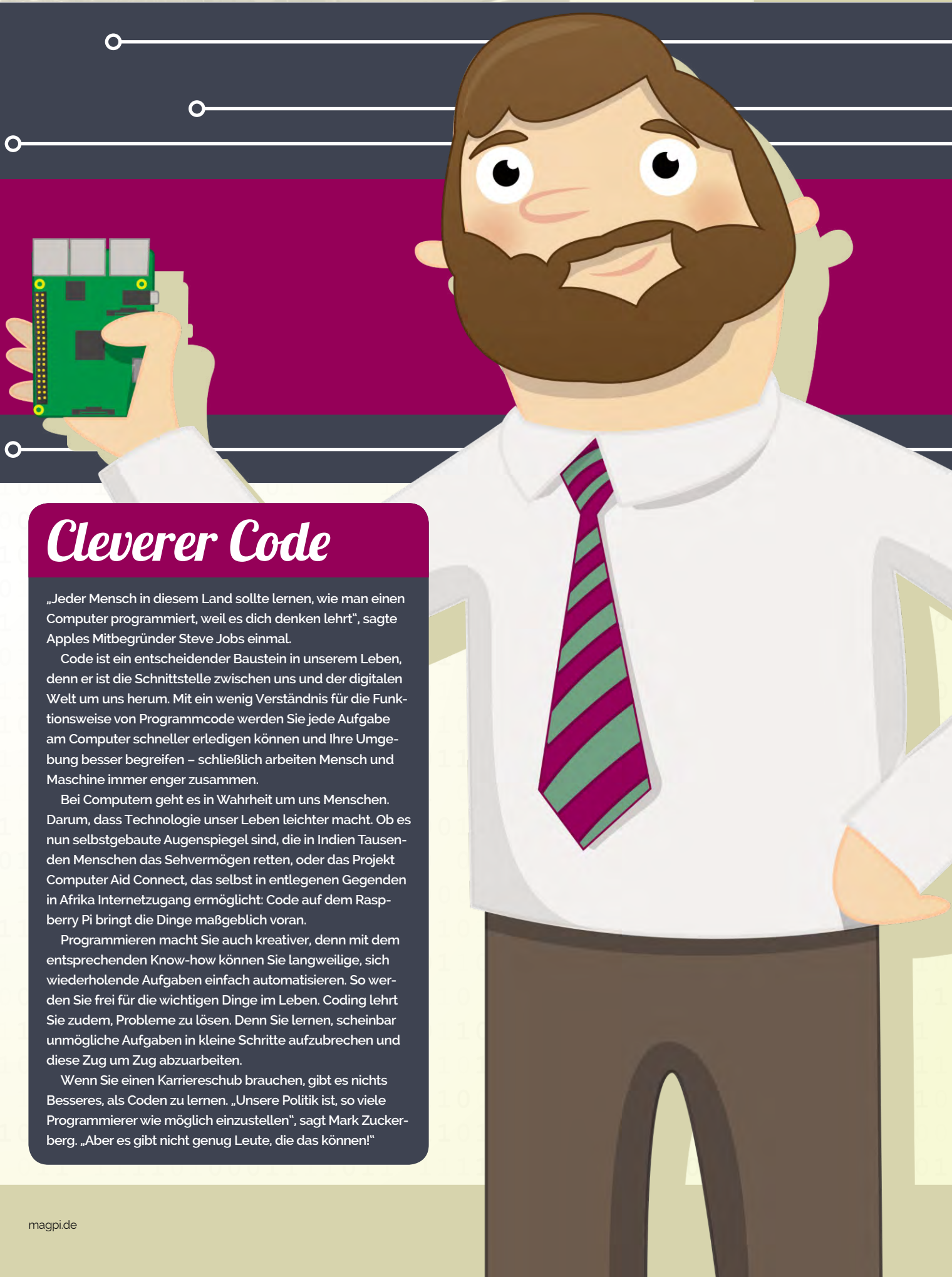
Programmieren zu lernen kann Ihr Leben nachhaltig verändern – und zwar zum Besseren, denn Coding wird in der modernen Welt immer wichtiger. Nur wenige elektronische Geräte kommen noch ohne die eine oder andere Zeile Code aus.

Einer der Gründe für die Entwicklung des Raspberry Pi war, dass an der Universität von Cambridge immer weniger Bewerbungen für den Studiengang Computerwissenschaften eingingen. Moderne Rechner und Spielekonsolen waren zu dieser Zeit zwar leistungsfähig und machten Spaß, waren aber nur schwer zu programmieren.

Die Community der Bastler verliebte sich sofort in den Raspberry Pi, da er günstig und modifizierbar ist. Mit etwas Kreativität und Tüftelei lassen sich eine Vielzahl von Projekten umsetzen. Am besten ist es natürlich, wenn Hard- und Software dabei eine fruchtbare Liaison miteinander eingehen.

Es ist egal, ob Sie ein fortgeschrittener Anwender oder ein Möchtegern-Programmierer mit dem Wunsch nach einem cooleren Job sind: In diesem Guide helfen wir Ihnen, Code besser zu verstehen. Die gute Nachricht: Sie müssen dazu kein Genie sein. Lesen und schreiben können Sie ja auch. Ähnlich einfach wird Ihnen das Programmieren von der Hand gehen, sobald Sie grundlegende Konzepte wie Variablen, Loops und Branching verstanden haben.

Vielleicht haben Sie früher mit BASIC oder älteren Sprachen wie Pascal oder Fortran gearbeitet. Aber egal, welche Vorkenntnisse Sie haben: Wir helfen Ihnen, die Geheimnisse des Codes zu entschleiern, damit Sie besser verstehen, was in Ihrem Raspberry Pi vor sich geht.



Cleverer Code

„Jeder Mensch in diesem Land sollte lernen, wie man einen Computer programmiert, weil es dich denken lehrt", sagte Apples Mitbegründer Steve Jobs einmal.

Code ist ein entscheidender Baustein in unserem Leben, denn er ist die Schnittstelle zwischen uns und der digitalen Welt um uns herum. Mit ein wenig Verständnis für die Funktionsweise von Programmcode werden Sie jede Aufgabe am Computer schneller erledigen können und Ihre Umgebung besser begreifen – schließlich arbeiten Mensch und Maschine immer enger zusammen.

Bei Computern geht es in Wahrheit um uns Menschen. Darum, dass Technologie unser Leben leichter macht. Ob es nun selbstgebaute Augenspiegel sind, die in Indien Tausenden Menschen das Sehvermögen retten, oder das Projekt Computer Aid Connect, das selbst in entlegenen Gegenden in Afrika Internetzugang ermöglicht: Code auf dem Raspberry Pi bringt die Dinge maßgeblich voran.

Programmieren macht Sie auch kreativer, denn mit dem entsprechenden Know-how können Sie langweilige, sich wiederholende Aufgaben einfach automatisieren. So werden Sie frei für die wichtigen Dinge im Leben. Coding lehrt Sie zudem, Probleme zu lösen. Denn Sie lernen, scheinbar unmögliche Aufgaben in kleine Schritte aufzubrechen und diese Zug um Zug abzuarbeiten.

Wenn Sie einen Karriereschub brauchen, gibt es nichts Besseres, als Coden zu lernen. „Unsere Politik ist, so viele Programmierer wie möglich einzustellen", sagt Mark Zuckerberg. „Aber es gibt nicht genug Leute, die das können!"

Was ist ein Programm?

Hier erfahren Sie, wie Software aufgebaut ist und was im Hintergrund passiert

Welches Python?

Python 2 und 3 werden am häufigsten genutzt. Version 3 ist die Zukunft, darum arbeiten wir damit. Viele Kurse unterrichten aber noch mit Version 2. Daher lohnt es sich, die Unterschiede zwischen beiden zu kennen: **magpi.cc/2gP6zX3**

Bevor Sie etwas überstürzen, sollten Sie wissen, was ein Programm überhaupt ist. Laut Lexikon handelt es sich dabei um „eine Folge von Anweisungen, um bestimmte Funktionen mithilfe eines Computers auszuführen." Programme sind fast so etwas wie Kochrezepte: Es gibt eine Liste von Zutaten, genannt „Variablen" und eine Liste von Arbeitsschritten oder „Funktionen". Sie folgen den Anweisungen aus dem Kochbuch und am Ende kommt dabei ein leckerer Kuchen heraus – hoffentlich. Computer haben dabei den Vorteil, dass sie dieselbe Schrittfolge immer wieder durchführen können. Eine Maschine kann also tausend Kuchen backen, ohne je müde zu werden – das würden Sie mit einem sogenannten „Loop" erreichen. Programme treffen aber auch Entscheidungen; sie können verschiedene Pfade begehen. Um beim Süßen zu bleiben: Entweder es gibt am Ende eine Schokotorte oder Donuts – je nachdem welche Variablen dem Computer zur Verfügung stehen.

Kompiliert vs. interpretiert

Python ist eine „interpretierte" Sprache. Sie schreiben den Code und führen das Programm aus. Unter der Haube wird es übersetzt und läuft dann. Sprachen wie C und Java sind „kompiliert". Sie schreiben das Programm, kompilieren den Code und erhalten eine Datei in Maschinensprache, die Sie dann ausführen.

Sie werden überrascht sein, wie wenig Vorwissen Sie brauchen, um mit dem Programmieren anzufangen. Ein paar Variablen und Funktionen – und Ihr Computer wird die ganze harte Arbeit von allein erledigen.

Im Inneren des RasPi

Im Herzen Ihres Raspberry Pi befinden sich Millarden von binären Ziffern, kurz Bits. Genauer gesagt 8.589.934.592 in dem einen Gigabyte RAM. All diese Ziffern können entweder auf 0 (low oder ausgeschaltet) oder 1 (high oder an) geschaltet sein. Alles, was Sie auf Ihrem Monitor sehen, was aus Ihren Lautsprechern kommt und Sie in Ihr Keyboard hacken, besteht aus diesen binären Informationen.

Da Menschen nicht wirklich in dieser Sprache mit Maschinen kommunizieren können, gibt es die Programmiersprachen als Schnittstelle. So werden für Menschen leicht verständliche Befehle wie `print()` eingegeben und anschließend in Maschinensprache übersetzt, die der Computer versteht und in die Tat umsetzen kann.

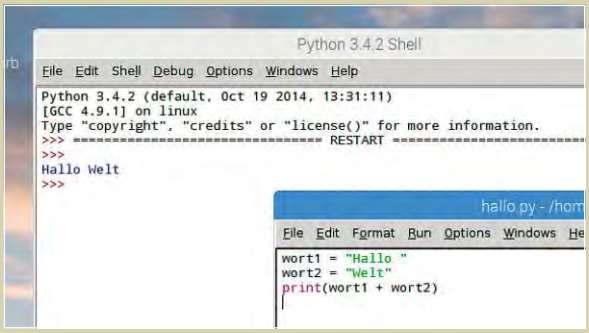
Wie bereits erwähnt geht es in diesem Workshop hauptsächlich um die Sprache Python. Sie ist großartig, da sie so einfach zu erlernen ist. Ihre Syntax ist so umfangreich wie aufgeräumt, Sie müssen sich keine Sorgen um geschweifte Klammern oder statische Typisierung machen, wie es in komplizierteren Sprachen à la Java der Fall ist. Mit Python schreiben Sie einfach Ihren Code, führen ihn aus und kriegen etwas gewuppt. Darum ist Python auch die Sprache, die in der **MagPi** am häufigsten verwendet wird – damit Sie die teils abgefahrenen Projekte leichter durchschauen und selbst umsetzen können.

IDE und IDLE

Sie müssen Python-Programme nicht in Texteditoren schreiben. Stattdessen gibt es praktische Komplettlösungen namens IDE (integrierte Entwicklungsumgebungen). Diese enthalten einen Editor, können den Code aber auch ausführen. Oft gibt es coole Funktionen wie Debugging oder Auto-Vervollständigung.

Klicken Sie auf **Menü | Entwicklung | Python 3 (IDLE)** und es öffnet sich ein neues Fenster namens „Python 3.4.2 Shell:“. Diese Shell funktioniert genau wie Python in der Kommandozeile. Geben Sie **`print("Hallo Welt")`** ein, um die Botschaft zu sehen. Im eingebauten Datei-Editor können Sie auch Programme schreiben. Klicken Sie auf **File | New File**. Geben Sie folgenden Code im leeren Fenster ein:

```
wort1 = "Hallo "
wort2 = "Welt"
print(wort1 + wort2)
```



Oben Python IDLE erleichtert das Erstellen und Testen von Programmen – ganz ohne Kommandozeile

Vergessen Sie das Leerzeichen nach „Hallo“ nicht. Speichern Sie die Datei nun als **hallo.py** und drücken Sie **F5** auf dem Keyboard, um das Programm auszuführen. In der Shell wird Ihnen nun der Text „Hallo Welt“ angezeigt. Der Vorteil von Python IDLE ist, dass Sie das Programm in der Shell direkt analysieren können. Geben Sie **`Wort1`** ein und Sie sehen „Hallo “, geben Sie **`Wort2`** ein und Sie sehen „Welt“. Dieses Feature macht das Experimentieren und die Fehlersuche im Code deutlich einfacher.

Warum Python?

Es gibt viele Programmiersprachen und jede hat ihre Vorzüge. Python ist ideal für Einsteiger. Denn die Syntax, also die verwendeten Zeichen und Befehle, ist leicht zu lesen. Doch mit Python lassen sich auch medizinische, wissenschaftliche oder Industrieprojekte umsetzen. So ist es auch für Profis bestens geeignet.

Python im Terminal

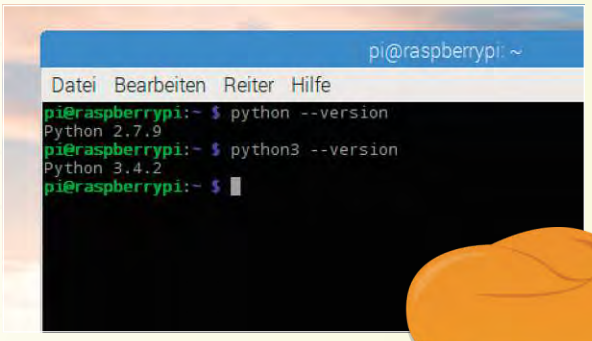
Sie müssen rein gar nichts tun, um Python auf dem Raspberry Pi einzurichten. Öffnen Sie einfach das Terminal und geben Sie **`python --version`** ein. Es wird „Python 2.7.9“ angezeigt. Geben Sie **`python3 --version`** ein und „Python 3.4.2“ erscheint. In diesem Artikel arbeiten wir ausschließlich mit Python 3 (siehe Kasten „Welches Python“). Starten Sie also Python, indem Sie einfach **`python3`** im Terminal eingeben.

Statt „$“ wird Ihnen nun „>>>“ angezeigt. Hier geben Sie Python-Befehle ein, genau wie Sie es im normalen Terminal machen. Es ist Tradition, jede neue Programmsprache mit „Hallo Welt“ einzuweihen. Geben Sie **`print("Hallo Welt")`** ein und drücken Sie [Enter]. In der nächsten Zeile wird Ihnen „Hallo Welt“ ausgegeben.

Die Shell ist auch als interaktiver Modus bekannt, da Sie so direkt mit dem Code interagieren können – etwa bei Matheaufgaben. Geben Sie **`1.920 * 1.080`** ein. Die Antwort lautet 2.073.600.

Die meiste Zeit werden Sie Python-Code aber mit normalen Texteditoren schreiben und die Dateien mit der Endung „.py“ speichern. Verwenden Sie aber keinen Office-Editor, da diese den Text formatieren und so den Code verwüsten.

Einfache Tools wie Leafpad genügen. Geben Sie den Code ein, speichern Sie ihn als Programm und führen Sie die Datei über das Terminal aus. Geben Sie dazu einfach **`python3 programmname.py`** ein.



Links Python ist bei Raspbian bereits vorinstalliert und kann direkt im Terminal genutzt werden

Variablen

… sind vielseitige Behälter zum Speichern von Daten und Objekten

Python-Typen

Python hat fünf Datentypen:

- Zahlen
- Strings
- Listen
- Tupel
- Dictionary

Foo bar?

Die Begriffe „foo“ und „bar“ werden Ihnen beim Programmieren oft begegnen. Es handelt sich dabei lediglich um Platzhalterbezeichnungen für Ihre Variablen. Genauso gut könnten diese auch zick und zack oder bim und bam heißen.

Wenn Sie bereits einmal mit Code experimentiert haben, sind Ihnen sicher schon Variablen begegnet. In der Wissenschaft handelt es sich dabei um jeden Faktor, den man kontrollieren oder messen kann. Beim Programmieren dienen Variablen aber dazu, Dinge aufzubewahren. Das können Namen, Zahlen oder Tags sein. Kurzum: alles, was Ihr Programm benötigt. In Python schreiben Sie zuerst den Namen der Variable, dann ein Gleichheitszeichen und schließlich ein Wort, eine Ziffer oder ein Objekt, das Sie dort hineinpacken möchten:

```
foo = 1
bar = 2
```

Denken Sie immer daran: Variablenname links, Inhalt rechts. Stellen Sie sich vor, Sie haben zwei Becher. Einer ist mit „foo“ und der andere mit „bar“ beschriftet. In den ersten packen Sie die Ziffer 1 und in den zweiten die Ziffer 2. Möchten Sie später nachsehen, welche Zahl sich worin befindet, schauen Sie einfach in die Becher. Dasselbe passiert, wenn Sie in Python den Namen der Variablen eingeben:

```
foo
bar
```

Sie können sich die Variablen auch mithilfe der Funktion **`print`** ausgeben lassen:

```
print(foo)
print(bar)
```

Variablen können auch Strings enthalten, also längere Zeichenketten in Form von Wörtern oder anderem Text. Eine String-Variable zu erstellen funktioniert fast so wie bei den Zahlen, nur dass Sie den Text mit einfachen (`' '`) oder doppelten Anführungszeichen (`" "`) einrahmen.

Letztere vereinfachen die Arbeit mit Apostrophen, wie beispielsweise bei **`print("Don't worry. Be Happy")`**. Diese Zeile würde schon nach ‘Don’ enden, wenn Sie einfache Anführungszeichen nutzen, also **`print('Don't worry, be happy')`** **`happy')`**. Gehen Sie auf Nummer sicher.

Warum Variablen wichtig sind

Variablen machen es deutlich leichter, Teile des Codes schnell zu ändern. Angenommen, Sie arbeiten bei der Märchenwald AG und haben diesen Code geschrieben:

```
print("Hänschen klein ging allein")
print("Hänschen klein ging allein")
print("Hänschen klein ging allein")
print("in die weite Welt hinein")
```

Doch dann kommt Ihr Chef mit den Ergebnissen der letzten Umfrage zu Ihnen: „Hänschen ist ein altbackener Name, ändern Sie das in Kevin, aber pronto!“

Nun müssten Sie den Namen in allen drei Zeilen von Hand ändern. Kein Problem. Haben Sie aber bei einem Projekt Tausende Zeilen Code geschrieben, müssten Sie eine ganze Woche arbeiten, um die Änderungen einzupflegen.

Dank der Variablen müssen Sie den Namen nur einmal definieren und können ihn dann überall in Ihrem

Code einsetzen. Hier das verbesserte Beispiel:

```
name = "Hänschen"

print(name + " klein ging allein)
print(name + " klein ging allein)
print(name + " klein ging allein)
print("in die Weite Welt hinein")
```

Am Ende wird derselbe Text ausgegeben wie beim ersten Beispiel. Kommen dann Änderungswünsche, müssen Sie den Namen nur an einer Stelle ändern:

```
name = "Dolly"
```

Dadurch wird das Gedicht an allen Stellen korrigiert.

Welcher Typ darf's sein?

Wenn Sie eine Variable in Python erstellen, wird ihr automatisch ein Typ zugewiesen, je nachdem was sie enthält. Prüfen können Sie den Typ einer Variable mit der Funktion **type()**. Geben Sie in der Shell ein:

```
foo = "zehn"
bar = 10
```

Nun verwenden Sie die Funktion **type()**, um die Typen der beiden Variablen zu prüfen:

```
type(foo)
type(bar)
```

Dort wird stehen **<class 'str'>** bei **foo** und **<class 'int'>** bei **bar**. Dieses Konzept zu verstehen ist wichtig, da verschiedene Typen von Variablen unterschiedlich miteinander arbeiten – und das nicht immer reibungslos. Fügen Sie etwa zwei Strings zusammen, werden diese kombiniert:

```
name = "Harry"
job = "Zauberer"
print("Du bist ein " + job + " " + name)
```

Die Nachricht „Du bist ein Zauberer Harry“ wird ausgegeben. Die beiden Strings wurden verkettet. Bei Zahlen sieht das jedoch schon wieder komplett anders aus. Hier ein Beispiel:

```
nummer1 = 6
nummer2 = 9

print(nummer1 + nummer2)
```

Statt die „6“ und die „9“ aneinanderzureihen, um eine „69“ zu erzeugen, berechnet Python die Summe und gibt Ihnen die Lösung „15“ aus.

Typumwandlung

Doch nun stellt sich die Frage, was passiert, wenn man einen String und eine Zahl kombiniert.

```
name = "Ben"
nummer = 10
print(name + nummer)
```

Das Programm spuckt eine Fehlermeldung aus: „TypeError: Can't convert 'int' object to str implicitly“. Das liegt daran, dass Python Strings und Zahlen nicht addieren kann, weil sie unterschiedlich funktionieren. Multiplikation wiederum klappt ganz gut:

```
print(name * number)
```

Dadurch erhalten Sie zehn Mal Ben aneinandergereiht: „BenBenBenBenBenBenBenBenBenBen“.

Möchten Sie aber „Ben10“ ausgeben lassen, müssen Sie die Zahl in einen String konvertieren. Dazu nutzen Sie die Funktion **str()** und schreiben die Zahl in die beiden Klammern. Speichern Sie dann das Ergebnis in einer neuen Variable namens **nummer_als_zahl**:

```
nummer_als_zahl = str(nummer)
print(name + nummer_als_zahl)
```

Dieser Code wird Ihnen den Text „Ben10“ ausgeben. Dieses Konzept nennt sich Typumwandlung oder Typecasting – also die Konvertierung einer Variable in einen anderen Typ. Sie können dank **int()** auch Strings in Zahlen verwandeln. Das ist praktisch, wenn Sie per **input()** von einem Nutzer eine Zahl erfahren möchten, die als String gespeichert wird. Hier sehen Sie ein kleines Programm zum Berechnen einer Potenz, wobei Basis und Exponent vom Nutzer bestimmt werden.

```
basis = input("Bitte Basis eingeben: ")
exponent = input("Exponent eingeben: ")
ergebnis = int(basis) ** int(exponent)
```

Die beiden Variablen **basis** und **exponent** sind Strings, aber die dritte, **ergebnis**, ist eine Zahl. Nun können Sie das Ergebnis anzeigen lassen:

```
print(ergebnis)
```

Möchten Sie eine Botschaft einschließen, müssen Sie **ergebnis** wiederum in einen String umwandeln:

```
print(basis + " hoch " + exponent +
"ergibt " + str(ergebnis))
```

Das klingt zunächst verwirrend. Doch Python ist gerade deshalb so einfach, weil es die Typen von Variablen dynamisch auf deren Inhalt anpasst. Dadurch ist allerdings auch etwas Vorsicht beim Coden nötig.

Wie sollten Variablen heißen?

Variablennamen sollten immer kleingeschrieben werden. Einzelne Wörter trennen Sie mit einem Unterstrich. Auch Zahlen sind möglich, allerdings muss die Variable mit einem Buchstaben beginnen. Ansonsten haben Sie freie Hand, abgesehen von einer Reihe feststehender Begriffe: **magpi.cc/2h7MH1y**. Von Vorteil sind treffende, eindeutige Namen, die zur Funktion passen (z. B. name_teilnehmer).

Alles im Griff dank While & For

Mit While- und For-Schleifen macht Ihr Programm die Arbeit von allein

Vergleichsoperatoren

Diese Operatoren werden stets bei Bedingungen eingesetzt, um festzulegen, ob etwas wahr („true“) oder falsch („false“) ist:

- `==` gleich
- `!=` nicht gleich
- `<` weniger als
- `<=` kleiner oder gleich
- `>` größer als
- `>=` größer oder gleich
- `<>` kleiner oder größer als

Computern ist es egal, ob sie dieselbe Aufgabe immer und immer wieder ausführen müssen. Genau das macht sie so praktisch, vor allem für repetitive Arbeiten. Im letzten Kapitel, als es um die Variablen ging, haben Sie den folgenden Text in Python per Shell ausgegeben:

```
print("Haenschen klein ging allein")
print("Haenschen klein ging allein")
print("Haenschen klein ging allein")
print("in die weite Welt hinein")
```

Haenschen haben wir mit einer Variable ersetzt, trotzdem war der Code noch amateurhaft. Denn warum sollten Sie dieselbe Zeile **`print`** drei Mal eingeben?

Setzen Sie stattdessen eine Schleife ein, um die Wiederholung auszuradieren. Die erste Schleife ist ein sogenannter „While-Loop“. Erstellen Sie in Python 3 IDLE eine neue Datei, zum Beispiel **hans.py** und geben Sie den Code oben rechts auf der nächsten Seite ein:

```
name = "Haenschen"
zaehler = 0
```

Dann nutzen Sie **`while`** gefolgt von einer Bedingung, nämlich **`zaehler < 3`**.

In der nächsten Zeile drücken Sie die Leertaste vier Mal, um den Code einzurücken. Benutzen Sie nicht die Tabulatortaste (siehe Kasten unten links).

```
while zaehler < 3:
    print(name + " klein ging allein")
    zaehler = zaehler + 1
```

Das Zeichen **`<`** steht für „kleiner als“. Es prüft, ob der Wert links geringer ist als der rechts. In diesem Beispiel, ob der Zähler, der bei Null beginnt, kleiner als drei ist. Dieser Zustand heißt „true“ (wahr), das Gegenteil heißt „false“ (falsch) und tritt dann ein, wenn der Zähler größer als drei wird.

Geben Sie schließlich die letzte Zeile ein:

```
print("in die weite Welt hinein")
```

Speichern Sie das Programm und starten Sie es mit der Taste **F5**. Dort sollte nun drei Mal „Haenschen klein ging allein“ stehen und danach „in die weite Welt hinein“.

While, Bedingungen, Einrückung

Hier sehen Sie drei wichtige Faktoren: Die While-Aussage, eine Bedingung (Condition), eingerückter Code:

```
while condition:
    eingerueckter Code
```

Stellen Sie sich das Ganze in Ihrem Programm **hans.py** als eine Art Dialog zwischen den Dreien vor:

Tabs oder Leerzeichen?

Die Nerds streiten sich online heftig darüber, ob man zum Einrücken von Zeilen Tabstopps oder Leerzeichen verwenden sollte. Es gibt Argumente für beide Seiten: Tabs sind schneller und erzeugen kleinere Dateien, Leerzeichen erlauben kleinere Einrückungen. Für den Anfang können Sie ruhig mit Letzterem arbeiten.

While: "Condition, wie lautet dein Status?"
Condition: "True! Zähler gleich 0, weniger als 3."
Eingerückt: "OK, Leute. Ich gebe ‚Haenschen klein ging allein' aus und erhöhe den Zähler um 1. Weiter."

While: "Condition, wie lautet dein Status?"
Condition: "True! Zähler gleich 1, weniger als 3."
Eingerückt: "OK. Ich gebe nochmal ‚Haenschen klein ging allein' aus und erhöhe den Zähler wieder um 1."

Das geht so weiter bis der Zähler 3 erreicht hat.

While: "Condition, wie lautet dein Status?"
Condition: "False! Der Zähler steht genau auf drei, das ist nicht weniger als drei."
While: "OK, Jungs, wir sind hier fertig!"

Nun wird nicht mehr der eingerückte Code ausgeführt, sondern es geht weiter mit „in die weite Welt hinein".

For und Listen

Die nächste Art von Schleifen heißt „for". Diese ist für die Arbeit mit Listen geeignet.

Listen sind eine Art von Variable, die mehrere Objekte wie Strings, Zahlen oder gar andere Variablen enthält. Listen erstellen Sie mit eckigen Klammern:

```
teletubbies = ["Tinkywinky", "Dipsy",
               "Lala", "Po"]
```

Geben Sie nun **teletubbies** in der Shell ein, um die Liste anzuzeigen. Ihnen werden die vier Namen nun in eckigen Klammern angezeigt. Sie können jedes der Objekte einzeln ansteuern. Der Code

```
teletubbies[0]
```

wird Ihnen „Tinkywinky" ausgeben. Listen in Python beginnen immer bei Null, also ist das erste Objekt immer die [0]. Demzufolge können Sie sich alle Namen mit diesem Code ausgeben lassen:

Hans.py

```
name = "Haenschen"
zaehler = 0

while zaehler < 3:
    print(name + " klein ging allein")
    zaehler = zaehler + 1

print("in die weite Welt hinein")
```

```
teletubbies[0] # "Tinkywinky"
teletubbies[1] # "Dipsy"
teletubbies[2] # "Lala"
teletubbies[3] # "Po"
```

Solche Listen können zunächst verwirrend sein. Ein For-Loop macht es einfacher, über Objekte einer Liste zu iterieren. Erstellen Sie das folgende Programm und speichern Sie es als **teletubbies.py**:

```
teletubbies = ["Tinkywinky", "Dipsy",
               "Lala", "Po"]

for teletubby in teletubbies:
    print(teletubby)
```

Es ist egal, wie Sie die Variable in einem For-Loop nennen, solange Sie sich an ihren Zweck erinnern. Sie können statt „teletubby" auch schreiben:

```
for bunter_vogel in teletubbies:
    print(bunter_vogel)
```

Es ist allerdings üblich, eine Liste mit einem Plural-Namen zu versehen (z. B. namen, strassen) und dessen Singular-Version als Variable zu verwenden, also „name in namen" oder „strasse in strassen" oder eben „teletubby in teletubbies".

Endlosschleifen

Sie müssen darauf achten, den Zähler in While-Loops zu ändern, sonst entsteht eine Endlosschleife. Löschen Sie etwa `zaehler = zaehler + 1` aus der Schleife, läuft das Programm für immer weiter, da der Wert nie über Null steigt. Diesen schwerwiegenden Bug sollten Sie in Ihrem Code um jeden Preis vermeiden.

Bedingte Anweisungen

So machen Sie Ihren Code wesentlich intelligenter

Logische Operatoren

Sie können Bedingungen mit logischen Operatoren kombinieren.

and Beide Operatoren sind true (a und b)

or Einer der Operatoren ist true (a oder b)

not Prüft, ob etwas false ist: not (a and b) tritt ein, wenn sowohl a als auch b false sind

Ihre Programme werden allmählich immer mächtiger. Sie haben gelernt, Anweisungen in fortlaufender Reihenfolge auszuführen und Ihren Code mit Variablen sowie Loops zu verbessern.

Ein weiterer wichtiger Bestandteil des Programmierens sind bedingte Anweisungen und Verzweigungen, im Englischen „Conditional Branching“. So kann ein Programm entscheiden, ob eine Aktion ausgeführt werden soll oder nicht. Das tut es aber nicht von allein, sondern nach einer strengen Logik.

Alles fängt mit dem mächtigen If-Befehl an. Dies ähnelt einem Loop, wird aber nur einmal ausgeführt. Dieses Statement prüft, ob eine Bedingung wahr ist (true). Falls ja, wird der Code ausgeführt:

```
if True:
    print("Hallo Welt")
```

Führen Sie das Programm aus, dann wird Ihnen der Text „Hallo Welt“ angezeigt. Ändern Sie es in false:

```
if False:
    print("Hallo Welt")
```

und es passiert: gar nichts. Natürlich – Sie können nicht einfach true und false schreiben. Stattdessen legen Sie Bedingungen fest, die an die beiden Zustände geknüpft sind. Ein übliches Beispiel dafür ist das Gleichheitszeichen (**==**). Es prüft, ob beide Objekte von ihm identisch sind. Erstellen Sie eine neue Datei und nennen Sie sie **passwort1.py**. Übernehmen Sie den Code oben von der nächsten Seite. Das Programm bittet Sie um die Eingabe eines Passworts. Ist dieses korrekt, wird der Text „Willkommen!“ angezeigt. Achten Sie darauf, den Operator **==** nicht mit dem normalen Gleichheitszeichen **=** zu verwechseln. Letzteres prüft nicht, ob beide Seiten gleich sind, sondern es sorgt dafür, dass sie es sind – das ist ein typischer Anfängerfehler unter Programmierern.

else als Alternative

Neben „If“ müssen Sie „else“ als Steuerungselement für Verzweigungen kennen. Die beiden gehen Hand in Hand, denn wenn „If“ wahr (true) ist, wird es ausgeführt. Ist es falsch (false), wird stattdessen „else“ ausgeführt, sozusagen als alternative Aktion.

```
if True:
   print("Der erste Zweig startet")
else:
   print("Der zweite Zweig startet")
```

Führen Sie diesen Code aus und es wird der Text „Der erste Zweig startet“ angezeigt. Der Code

```
if False:
   print("Der erste Zweig startet")
else:
   print("Der zweite Zweig startet")
```

zeigt Ihnen dann „Der zweite Zweig startet“ an. Verbessern Sie damit das Passwort-Programm (oben rechts) und speichern Sie die Datei **passwort2.py**. Auch hier werden Sie mit „Willkommen!“ begrüßt, sollte das Passwort stimmen. Falls nicht, spuckt der Code die Botschaft „Falsches Passwort!“ aus.

Elif (else if)

Die dritte Größe, die Sie kennen müssen, heißt „elif“. Das steht für „else if“ und befindet sich demnach genau zwischen if und else. Geben Sie zum Beispiel den folgenden Code ein:

```
if False:
    print("Der erste Zweig startet")
elif True:
    print("Der zweite Zweig startet")
else:
    print("Der dritte Zweig startet")
```

Wenn Sie dieses Programm ausführen, wird das If-Statement direkt übersprungen. Stattdessen wird elif ausgeführt und dessen Text wird angezeigt.

Das else-Statement ist nicht true oder false, es kommt nur dann zum Einsatz, wenn sowohl if als auch elif false sind. Beachten Sie, dass else wie immer optional ist. Sie können auch nur mit if und elif arbeiten.

Was passiert aber, wenn Sie if und elif auf true setzen? Versuchen Sie es einmal selbst und experimentieren Sie ein wenig. Dadurch bekommen Sie ein besseres Verständnis für die Funktionsweisen dieser drei Operatoren für bedingte Anweisungen.

FizzBuzz

Wir zeigen Ihnen nun ein typisches Programm namens „FizzBuzz“. Damit können Sie testen, ob Sie die Statements if, else und elif wirklich verstehen.

Zuerst müssen Sie den Operator namens Modulo kennen (**%**). Er wird verwendet, um die Reste einer Division zu ermitteln. Darum ähnelt er auch dem Operator fürs Dividieren:

```
10 / 4 == 2.5
```

Mit einem Modulo bekommen Sie stattdessen:

```
10 % 4 == 2
```

Der Modulo kann durchaus praktisch sein, etwa um zu ermitteln, ob eine Zahl gerade oder ungerade ist:

```
10 % 2 == 0 # das ist gerade
11 % 2 == 1 # das ist ungerade
```

Das klappt mit geraden wie ungeraden Zahlen:

```
nummer = 10

if nummer % 2 == 0:
    print("Die Zahl ist gerade")
else:
    print("Die Zahl ist ungerade")
```

Weiter gehts mit FizzBuzz.

Passwort1.py

```
passwort = "qwertz"
versuch = input("Passwort eingeben: ")

if versuch == passwort:
    print("Willkommen!")
```

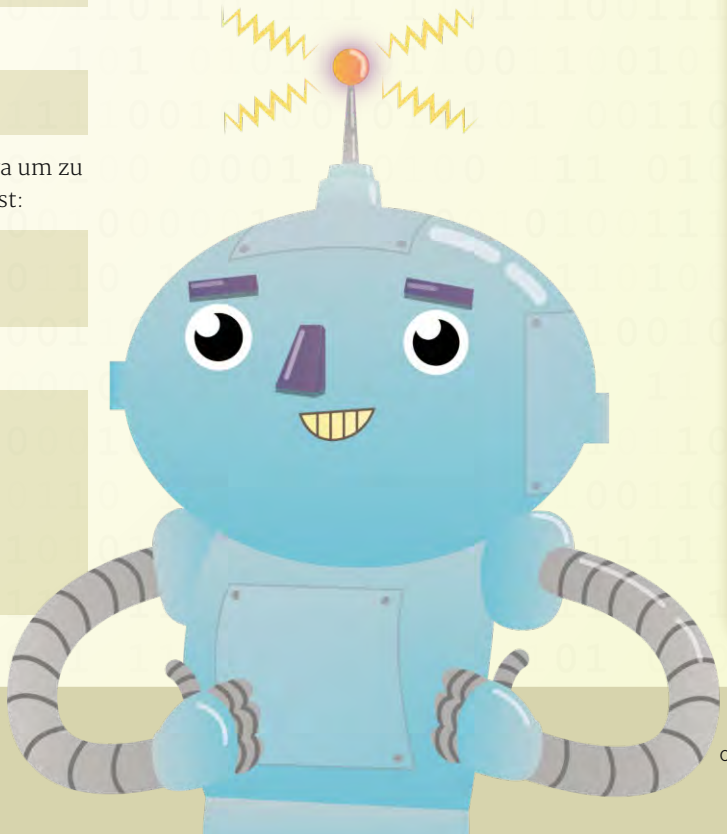
Passwort2.py

```
passwort = "qwertz"
versuch = input("Passwort eingeben: ")

if versuch == passwort:
    print("Willkommen!")
else:
    print("Falsches Password!")
```

FizzBuzz ausgeben

Unser FizzBuzz soll alle Zahlen von eins bis 100 ausgeben. Ist eine Ziffer durch drei teilbar, soll das Programm stattdessen „Fizz“ ausgeben. Ist sie teilbar durch fünf, soll es „Buzz“ anstelle der Zahl ausgeben. Ist eine Zahl, etwa 15, durch beide Zahlen teilbar, soll das Programm „FizzBuzz“ anzeigen. An dieser Stelle führen wir auch gleich ein neues Element ein: Das And-Statement. Es prüft, ob beide Bedingungen true sind: Nämlich ob eine Zahl sowohl durch drei als auch durch fünf teilbar ist. Diese Funktion gibt nur ein true aus, wenn beide Bedingungen erfüllt werden.

Es gibt drei logische Operatoren: „and“, „or“ sowie „not“. Die ersten beiden sind selbsterklärend, aber „not“ ist zunächst verwirrend. Mit etwas Übung gibt sich das aber schnell. Geben Sie zum Beispiel den Code aus **fizzbuzz.py** von Seite 27 ein, um den Umgang mit if, else und elif sowie den logischen Operatoren zu üben.



Tipp

Ein guter Programmierer kommentiert seinen Code. So wird jeder Baustein in kurzen Sätzen für Menschen verständlich gemacht. Der Computer ignoriert diese Kommentare. In Python setzen Sie einfach eine Raute vor den Text. Der Kommentar kann hinter Code stehen oder eine Zeile für sich sein. Alles, was hinter # steht, wird von Python nicht in Maschinensprache übersetzt.

So ticken Funktionen

Erstellen Sie blockweise Code und bauen Sie so stabile Programme

Sie haben sich gut geschlagen seit Ihrem ersten „Hallo Welt!“. Ihre Programme können nun Bedingungen abklopfen und Schleifen fahren. Sie können also Programme verfassen, die man als „Turing-vollständig“ bezeichnet – benannt nach Alan Turing, dem Paten der Computerwissenschaften, der im Zweiten Weltkrieg den Enigma-Code der Nationalsozialisten knackte. Lassen Sie uns jetzt noch einen Gang hochschalten.

Wir weihen Sie nun in die spannende Welt der Funktionen ein. Es handelt sich dabei um ganze Blocks an Code, die Sie nur einmal schreiben und überall beliebig oft wiederholen können.

Funktionen erkennen

Python ist vollgepackt mit eingebauten Funktionen und Sie haben sogar schon welche benutzt, nämlich **`print()`**, **`len()`** oder **`type()`**. Funktionen sind also ziemlich leicht zu erkennen: Sie beginnen alle mit mindestens einem Kleinbuchstaben gefolgt von zwei runden Klammern.

Python-Dokumentation

Sie können die ausführliche Python-Dokumentation direkt auf der Website **python.org/doc** ansehen oder downloaden – allerdings nur auf Englisch. Auf der Seite **docs.python.org/2/library/functions.html** werden viele der eingebauten Funktionen von Python erklärt und übersichtlich aufgelistet.

Funktionen einsetzen

Schauen Sie sich einmal die Funktion **`abs()`** an. Das steht für „absolute“ und gibt Ihnen den absoluten Wert jeder Zahl aus, die Sie eingeben. Der Wert in den Klammern heißt „Argument“.

Eine absolute Zahl ist die positive Version derselben Zahl. Probieren Sie dies einmal in der Shell aus:

```
abs(2) # ergibt 2
abs(-2) # ergibt 2
```

Sie können diesen Wert als Variable speichern:

```
positive_nummer = abs(-10)
```

Oft ist es einfacher, Funktionen von rechts nach links zu lesen, also: Der Wert in den Klammern wird in eine absolute Zahl verwandelt und dann in einer neuen Variable gesichert.

Funktionen definieren

Das Tolle an Python ist, dass Sie nicht nur die eingebauten Funktionen nutzen können. Sie können auch benutzerdefinierte Funktionen anlegen.

Das gelingt Ihnen mit dem Begriff **`def`**, gefolgt vom Namen der Funktion und Klammern. In den Klammern listen Sie wieder die Parameter auf. Ein Parameter ist prinzipiell dasselbe wie ein Argument, doch innerhalb der Definition heißen sie eben anders.

```
def function(parameter):
    parameter ausgeben
```

Diese Funktion kann eigentlich nichts. Sie nimmt nur einen Parameter an und gibt ihn wieder aus.

Am Ende jeder Funktions-Definition steht ein Doppelpunkt (**:**). Jede Funktion wird mit vier Leerzeichen eingerückt, genau wie ein Loop oder if/else.

Dieser Code wird ausgeführt, sobald Sie die Funktion ansteuern. Funktionen enthalten typischerweies einen Befehl **return**, der eine Antwort ausgibt.

Funktionierende Funktionen

Als Beispiel erstellen wir eine Funktion, die den Text von „Happy Birthday“ anzeigt.

Schreiben Sie den Code aus **happy_birthday.py** in eine Datei und führen Sie den Code aus.

```
happy_birthday("Thorsten")
```

Dieser Aufruf der Funktion verwendet den String „Thorsten“ als Argument. Dieser String wird an die Funktion als Parameter weitergegeben und wird für den vorgesehenen Code verwendet.

Return-Anweisungen

Viele Funktionen führen nicht nur Code aus, sondern antworten auch, wenn sie aufgerufen werden.

Im Beispiel mit **abs()** wurde die absolute Zahl einer beliebigen Nummer ausgegeben, die dann als Variable gespeichert wurde. Diese Funktion rekonstruieren wir hier, damit Sie sehen, was hinter den Kulissen passiert. In der Mathematik werden Zahlen durch Multiplikation mit -1 invertiert, etwa:

```
10 * -1 = -10
-10 * -1 = 10
```

Sie müssen nun eine Funktion erstellen, die eine Zahl als Parameter nutzt und prüft, ob diese negativ ist. Falls ja, wird sie mit -1 multipliziert. Falls nein, wird einfach der Parameter ausgegeben. Wir nennen die Funktion **absolute()**. Den Code finden Sie oben unter **absolute.py**. Erreicht die Funktion einen der **return** Befehle, wird der entsprechende Wert ausgegeben und die Funktion beendet.

Führen Sie den Code in **absolute.py** aus und geben Sie in der Shell Folgendes ein:

```
absolute(10)
absolute(-10)
```

Das letzte Programm „FizzBuzz“ haben wir auf Seite 25 schon erwähnt – es hilft Ihnen dabei, if, else und elif zu verstehen. Außerdem sollten Sie den Operator namens Modulo kennen, um „FizzBuzz“ zum Laufen zu kriegen. Gut beschrieben ist er hier: **python4kids.net/how2think/kap04.htm** (auf Deutsch). Er gibt per **return** den Rest einer Division aus. Nun können Sie sich durch den Code oben unter **fizzbuzz.py** arbeiten.

Happy_birthday.py

```
def happy_birthday(name):
    zaehler = 0
    while zaehler < 4:
        if zaehler != 2:
            print("Happy birthday to you")
        else:
            print("Happy birthday lieber " + name)
        zaehler += 1
```

Absolute.py

```
def absolute(number):
    if nummer < 0:
        return nummer * -1
    else:
        return nummer
```

Fizzbuzz.py

```
zaehler = 0
ende = 100

while zaehler < ende:
    if zaehler % 5 == 0 and zaehler % 3 == 0:
        print("FizzBuzz")
    elif zaehler % 3 == 0:
        print("Fizz")
    elif zaehler % 5 == 0:
        print("Buzz")
    else:
        print(zaehler)

    zaehler += 1
```

Weitere Informationen

Hier zeigen wir Anlaufstellen, die Ihnen sicher helfen:

GPIO Zero Essentials – magpi.cc/2bA3ZP7
Dieses Handbuch zeigt Ihnen die Grundlagen und erklärt Ihnen, welche Features die Python-Module des GPIO Zero bieten.

FutureLearn – magpi.cc/2h5Sthf
Die Raspberry Pi Foundation hat zwei neue Onlinekurse im Angebot. Dort lernen Sie viel über Physical Computing mit dem RasPi und Python. Außerdem gibt es einen Programmierkurs für Grundschulen.

Python Tutorial auf Youtube – bit.ly/2kqOKyn
In über 30 deutschsprachigen Video-Workshops wird anschaulich in die Programmierung eingeführt.

Learning Python – magpi.cc/2h2opWC
Dieses Tutorial von der Raspberry Pi Foundation bringt Dateien mit, die Sie direkt herunterladen können.

Import von Code

Bauen Sie auf der Arbeit erfahrener Coder auf, um noch bessere Resultate zu erzielen

Pygame

Wenn Sie mehr über Pygame erfahren möchten, schauen Sie sich diesen kostenlosen, englischen Grundlagen-Guide zu dem Modul an. **magpi.cc/2h2m0vh**

Natürlich erwartet niemand von Ihnen, dass Sie das Rad neu erfinden und den kompletten Code für Ihre Projekte neu schreiben. Stattdessen können Sie in vielen Fällen auf den Vorleistungen anderer Programmierer aufbauen. Das Grundgerüst steht dann im Nu.

Ihre Programme können nämlich mit dem Befehl **import** auch eine Menge Code von anderswo importieren. Ganze Module und deren Funktionen holen Sie so ins Boot – nur dass man diese dann „Methoden" nennt. Module importieren Sie in der Kommandozeile. Auf die Funktionen weisen Sie mit einem kleinen Punkt hin. Sie listen also zuerst das Modul, dann einen Punkt (**.**) und schließlich die Methode auf. Typisches Beispiel: **math**. Damit können Sie viele Rechenaufgaben erledigen. Öffnen Sie eine Python-Shell:

```
import math
```

Nun greifen Sie auf alle Methoden aus **math** zu. Sie werden keinen Unterschied merken. Geben Sie

```
type(math)
```

ein und es wird <class 'module'>‘ anzeigen. Geben Sie probehalber einmal **math**, einen Punkt und dann eine beliebige Methode ein, die Sie testen wollen:

```
math.sqrt(16)
```

Dieser Code gibt Ihnen die Quadratwurzel von 16 aus, also 4. Manche Methoden besitzen mehrere Argumente. **math.pow()** etwa berechnet Potenzen:

```
math.pow(64,3)
```

Das korrekte Ergebnis lautet: 262.144,0.
Sie können auch auf konstante Werte eines Moduls zugreifen, also fixe Variablen innerhalb des Moduls. Diese sehen aus wie normale Funktionen, nur ohne die runden Klammern.

```
math.pi
```

Dieser Code gibt Ihnen die Kreiszahl Pi auf 15 Stellen hinter dem Komma genau aus: 3.141592653589793.

```
math.e
```

Dieser wiederum zeigt Ihnen die Eulersche Zahl auf 15 Stellen hinter dem Komma an: 2,718281828459045.

Sie können mit **from** auch Methoden und Konstanten von anderen Modulen importieren. Dadurch können Sie diese auch innerhalb Ihres Programms nutzen, ohne einen Punkt verwenden zu müssen:

```
from math import pi
from math import e
from math import pow
```

Wenn Sie nun irgendwo **pi** oder **e** eingeben, werden Ihnen die entsprechenden Zahlen sofort angezeigt. Sie können auch **pow()** wie eine normale Funktion nutzen. Ändern Sie einfach ihren Namen mit **as**:

```
from math import pi as p
```

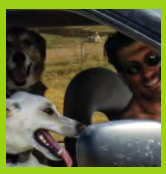
Nun genügt es, wenn Sie **p** eingeben und Ihnen wird Pi angezeigt. Übertreiben Sie es mit dem Umbenennen der Funktionen mithilfe von **as** allerdings nicht. Das kann verwirrend werden. Indem Sie eigene Funktionen erfinden und die von anderen Programmierern in Ihren Code importieren, lassen sich die Möglichkeiten Ihrer Programme um ein Vielfaches erweitern.

Sie können nun alles, was Sie im Einsteigerkurs gelernt haben, in die Tat umsetzen: Erstellen Sie etwa den Spieleklassiker Pong in Python. Legen Sie die Datei **pong.py** an und schreiben Sie den Code auf der rechten Seite sorgfältig ab. Oder verwenden Sie die Datei auf der Heft-DVD. Sie basiert auf der erweiterten Version von Trever Appleton (**magpi.cc/2hgkOUX**, ebenfalls auf DVD). Darin finden Sie Variablen, Funktionen, Loops und bedingte Anweisungen – also alles, was Sie in diesem Einsteigerworkshop gesehen haben. Trever Appletons Version besitzt sogar einen Punktezähler und fortgeschrittenen Code. Noch viel wichtiger aber ist, dass er auf seinem Blog Schritt für Schritt durch den Code führt (auf Englisch).

Wir hoffen, dass der Crashkurs für Sie der Anfang einer erfolgreichen Laufbahn als Programmierer sein wird – es gibt da draußen viel zu entdecken!

Pong.py

```
01. import pygame, sys
02. from pygame.locals import *
03.
04. # Variablen für das Spiel festlegen
05. window_width = 400
06. window_height = 300
07. line_thickness = 10
08. paddle_size = 50 # kleinere Plattform = schwereres Spiel
09. paddle_offset = 20
10.
11. # farbige Variablen werden erstellt
12. black = (0  ,0  ,0  ) # Variablen in Klammern heißen „Tupel“
13. white = (255,255,255) # Tupel sind wie Listen, doch die Werte blei-
    ben immer gleich
14.
15. # Variablen für die Bälle (x, y Cartesian coordinates)
16. # Startposition entspricht der Mitte der Arena
17. ballX = window_width/2 - line_thickness/2
18. ballY = window_height/2 - line_thickness/2
19.
20. # Variablen zum Messen der Richtung des Balls
21. ballDirX = -1 ## -1 = left 1 = right
22. ballDirY = -1 ## -1 = up 1 = down
23.
24. # Startposition in der Mitte der Arena
25. playerOnePosition = (window_height - paddle_size) /2
26. playerTwoPosition = (window_height - paddle_size) /2
27.
28. # Rechtecke für Plattformen und Ball erstellen
29. paddle1 = pygame.Rect(paddle_offset,playerOnePosition, line_
    thickness,paddle_size)
30. paddle2 = pygame.Rect(window_width - paddle_offset - line_thick-
    ness, playerTwoPosition, line_thickness,paddle_size)
31. ball = pygame.Rect(ballX, ballY, line_thickness, line_thickness)
32.
33. # Arena wird aufgemalt
34. def drawArena():
35.     screen.fill((0,0,0))
36.     # Außengrenze der Arena
37.     pygame.draw.rect(screen, white, (
    (0,0),(window_width,window_height)), line_thickness*2)
38.     # Mittellinie der Arena
39.     pygame.draw.line(screen, white, (
    (int(window_width/2)),0),((int(window_width/2)),window_height), (
    int(line_thickness/4)))
40.
41. # Plattformen werden gemalt
42. def drawPaddle(paddle):
43.     # Plattformen dürfen nicht zu weit nach unten
44.     if paddle.bottom > window_height - line_thickness:
45.         paddle.bottom = window_height- line_thickness
46.     # Plattformen dürfen nicht zu weit nach oben
47.     elif paddle.top < line_thickness:
48.         paddle.top = line_thickness
49.     # Plattform wird gemalt
50.     pygame.draw.rect(screen, white, paddle)
51.
52. # Funktion zum Malen des Balls
53. def drawBall(ball):
54.     pygame.draw.rect(screen, white, ball)
55.
56. # Funktion zum Bewegen des Balls
57. def moveBall(ball, ballDirX, ballDirY):
58.     ball.x += ballDirX
59.     ball.y += ballDirY
60.     return ball # returns new position
61.
62. # Funktion prüft Kollision des Balls mit der Wand und ändert dessen
    Richtung
63. def checkEdgeCollision(ball, ballDirX, ballDirY):
64.     if ball.top == (line_thickness) or ball.bottom == (window_
    height - line_thickness):
65.         ballDirY = ballDirY * -1
66.     if ball.left == (line_thickness) or
    ball.right == (window_width - line_
67.     thickness):
68.         ballDirX = ballDirX * -1
69.     return ballDirX, ballDirY # neue Richtung
70.
71. # Funktion prüft, ob der Ball eine Plattform trifft
72. def checkHitBall(ball, paddle1, paddle2, ballDirX):
    if ballDirX == -1 and paddle1.right == ball.left and
73. paddle1.top < ball.top and paddle1.bottom > ball.bottom:
74.         return -1 # neue Richtung (rechts)
    elif ballDirX == 1 and paddle2.left == ball.right and
75. paddle2.top < ball.top and paddle2.bottom > ball.bottom:
76.         return -1 # neue Richtung (rechts)
77.     else:
78.         return 1 # neue Richtung (links)
79.
80. # Funktion für Computergegner
81. def artificialIntelligence(ball, ballDirX, paddle2):
82.     # bewegt sich der Ball weg von ihm, geht er in die Mitte
83.     if ballDirX == -1:
84.         if paddle2.centery < (window_height/2):
85.             paddle2.y += 1
86.         elif paddle2.centery > (window_height/2):
87.             paddle2.y -= 1
88.     # kommt der Ball näher wird die Bewegung verfolgt
89.     elif ballDirX == 1:
90.         if paddle2.centery < ball.centery:
91.             paddle2.y += 1
92.         else:
93.             paddle2.y -=1
94.     return paddle2
95.
96. # Spielfenster initialisieren
97. screen = pygame.display.set_mode((window_width,window_height))
98. pygame.display.set_caption('Pong') # Wir im Fenster angezeigt
99.
100. # Arena und Plattformen malen
101. drawArena()
102. drawPaddle(paddle1)
103. drawPaddle(paddle2)
104. drawBall(ball)
105.
106. # Cursor unsichtbar machen
107. pygame.mouse.set_visible(0)
108.
109. # Das Hauptspiel läuft in diesem Loop
110. while True: # Endlosschleife. Mit Strg+C wird dsa Spiel beendet
111.     for event in pygame.event.get():
112.         if event.type == QUIT:
113.             pygame.quit()
114.             sys.exit()
115.         # Mausbewegungen
116.         elif event.type == MOUSEMOTION:
117.             mousex, mousey = event.pos
118.             paddle1.y = mousey
119.
120.     drawArena()
121.     drawPaddle(paddle1)
122.     drawPaddle(paddle2)
123.     drawBall(ball)
124.
125.     ball = moveBall(ball, ballDirX, ballDirY)
        ballDirX, ballDirY = checkEdgeCollision(
126. ball, ballDirX, ballDirY)
        ballDirX = ballDirX * checkHitBall(
127. ball, paddle1, paddle2, ballDirX)
128.     paddle2 = artificialIntelligence (ball, ballDirX, paddle2)
129.     pygame.display.update()
```



BRUCE SHAPIRO

Bruce begeisterte sich schon in seiner Jugend für Musik, Elektronik und fürs Basteln. Seine große Leidenschaft ist die Bewegungssteuerung.
magpi.cc/2fUUaRN

Im Innern des Tisches sitzt ein Raspberry Pi

Unter der Sandfläche sitzt ein Roboter, der Sisbot. Er steuert mithilfe eines starken Magneten die Kugel

Eine Metallkugel bewegt sich durch die Sandfläche und erzeugt die Muster

Infos

- Die Tische werden in den USA hergestellt
- Bruce kreiert schon seit 20 Jahren Sisyphus-Tische
- Sie werden in Museen und Galerien ausgestellt, unter anderem in Deutschland und der Schweiz
- Die Komponenten werden danach ausgesucht, dass sie möglichst ruhig laufen

SISYPHUS

Dieser Tisch ist ein richtiges Kunstwerk – und ein RasPi-Roboter

Bruce Shapiro ist gleichermaßen Bastler und Künstler. Anders als Picasso oder Rembrandt malt Bruce aber nicht in Öl auf Leinwand. „Ich beschäftige mich mit Bewegungssteuerung“, erläutert er uns.

Wir haben uns mit Bruce getroffen, um mit ihm über sein Projekt Sisyphus zu sprechen. Bruce hat gerade eine phänomenal erfolgreiche Kickstarter-Kampagne abgeschlosssen und hat uns eine Menge zu erzählen.

„Sisyphus ist ein computergesteuertes Gerät, das einen Magneten unterhalb einer Sandfläche entlangführt“, erklärt Bruce. „Auf dem Sand befindet sich eine Stahlkugel, die den Bewegungen des Magneten folgt und dadurch Muster in den Sand zeichnet.“

In der griechischen Mythologie war Sisyphos dazu verdammt, einen Felsbrocken auf einen Berg hinaufzuwälzen - immer und immer wieder, bis in alle Ewigkeit. „In meiner künstlerischen Interpretation ist Sisyphus eine kinetische Skulptur, die eine Kugel durch den Sand bewegt und immerwährend wunderschöne Muster zeichnet und wieder löscht. Sisyphus zu betrachten, hat etwas ausgesprochen Meditatives“, so Bruce.
Bruce Shapiro stellt seine Sisy-

Oben Der Tisch kann auch mit mehreren Kugeln ausgestattet werden

phus-Skulpturen schon seit fast 20 Jahren her und hat Dauerausstellungen in der Schweiz, in Deutschland und Australien. Kernstück des Projekts ist der sogenannte Sisbot, ein Roboter, der die Metallkugel steuert, die das Kunstwerk in den Sand „schreibt".

„Sisyphus ist eine CNC-Fräse", erläutert Bruce. „Sie nutzt jedoch keinen G-Code als Dateiformat. Das Prinzip ist aber das gleiche: Eine Fräsbahn entscheidet, wohin und in welcher Geschwindigkeit die Kugel sich bewegt. Nachdem der Sisbot polar arbeitet, führen die Bewegungen zu spiralförmigen Bögen. „Meine Muster sind eher algorithmisch, da ich nie gelernt habe zu zeichnen", fährt Bruce fort. „Aber eigentlich kann jeder Bahnen für Sisyphus erzeugen. Man muss einfach etwas malen, ohne den Stift abzusetzen. Wenn Sie dazu in der Lage sind, die Bewegungen Ihres Stiftes beim Zeichnen im Kopf zu behalten, können Sie mit Sisyphus arbeiten."

Ein Raspberry Pi eignet sich perfekt dafür, den Sisbot zu steuern und die Kunstwerke zu erzeugen – aber das war nicht immer so. „Lange Zeit wurden meine bewegungsgesteuerten Kunstwerke von einem Windows-PC angetrieben", meint Bruce. Drei davon tun es sogar immer noch – sie laufen Tag für Tag im Museum. „Ich verändere ungern etwas an Dingen, die eigentlich gut laufen", gibt Bruce zu. „Aber nach vielem Herumprobieren – das manchmal in der Sackgasse endete – habe ich die Erfahrung gemacht, dass eine Community sehr wichtig ist."

> „Ein Pi ist perfekt dazu geeignet, die Kunstwerke zu steuern"

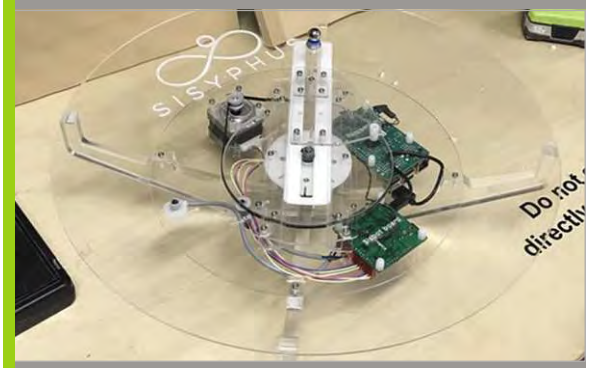
Folglich war es die Raspberry-Pi-Community, die Bruce davon überzeugte, zu dem supergünstigen Minicomputer zu wechseln. „Die Community war wichtiger für die Entscheidung als der Formfaktor und die geringen Kosten", erzählt er. „Die Entscheidung, den Raspberry Pi in der neuen Home-

SO FUNKTIONIERT DAS KUNSTWERK

>SCHRITT-01

Der Sisbot

Unter dem Tisch sitzt ein Roboter mit zwei Motoren, der Sisbot. Er steuert einen Magneten, der seinerseits die Stahlkugel durch den Sand bewegt.



>SCHRITT-02

Bewegungsabläufe

Die Motoren von Sisyphus werden von einem Pi gesteuert. Dieser generiert eine Reihe von Bahnen, die dann von der Kugel umgesetzt werden.

>SCHRITT-03

Immer in Aktion

Sisyphus hat keinen Ein-/Ausschalter. Wenn er eingesteckt ist, fährt er hoch und läuft. Sie können ihn übers WLAN ansteuern, etwa per iPhone-App. Über die App lässt sich die Geschwindigkeit des Sisbot und die Beleuchtung des Tisches steuern.

Die Metallkugel folgt einer Bahn, die von einer CNC-ähnlichen Technik vorgegeben wird

version von Sisyphus einzusetzen, verdanke ich letztlich den Thirty-somethings, mit denen ich mich austausche“, so Bruce. Die Community überzeugte ihn, dass JavaScript essenziell ist und dass Node.js auf einem Pi gut läuft.

„Die große Community und deren Bereitschaft, Wissen zu teilen, macht den Pi zu etwas Einzigartigem“, ergänzt Bruce.

Die Anforderungen an den Sisyphus-Tisch sind hochkomplex. „Das Ganze läuft auf drei Ebenen“, erklärt Bruce. Auf der untersten Ebene läuft eine in C geschriebene Firmware, die auf dem SisBot-Board läuft. Das Programm wurde von Brian Schmalz geschrieben (**magpi.cc/2fXxWhl**). Ursprünglich diente es einem früheren Projekt von Bruce, dem EggBot. „Brian startete vor vielen Jahren mit dem rudimentären Code, den ich für meine ersten Projekte geschrieben habe“, meint Bruce. Die zweite Ebene besteht aus einem Programm zur Bewegungssteuerung, das vor Kurzem in JavaScript portiert wurde und von Alex Wayne (**magpi.cc/2fXDqs7**) gründlich bereinigt wurde. Dieses Programm läuft in Node auf dem Pi. Auf der dritten Ebene werden die algorith-

mischen Bahnen erzeugt. „Ursprünglich erledigte ich das mithilfe von AutoLISP-Routinen, die in einem betagten AutoCAD liefen“, sagt Bruce, „aber inzwischen nutze ich das Grasshopper Rhino 3D plug-in.“

Wie alles begann

„Schon vor 25 Jahren faszinierte mich die Idee, meine Computer mit Motoren zu verbinden“, erinnert sich Bruce. „Um zu zeigen, was für ein Potenzial sich dahinter verbirgt, habe ich eine Maschine entwickelt, die zeichnen kann und dafür lediglich zwei Motoren benötigt.“ EggBot war die erste „Kunstmaschine“, die Bruce entworfen hat. „[EggBot] ist ziemlich cool, und die meisten Leute, denen ich es gezeigt habe, finden das auch“, meint er. „Ich habe viele Jahre damit verbracht, immer größere Zeichenmaschinen zu entwickeln, die dazu in der Lage waren, Schneidetools wie etwa Plasmabrenner zu bewegen. Damit konnte ich komplizierte Formen schneiden, um Skulpuren zu erschaffen – meist aus Metall. Als ich im Lauf der Jahre immer besser darin wurde, neue Geräte zu konstruieren, wurden auch die Komponenten, die ich dafür benötigte – Schrittmotoren und die dazugehörige Elektronik – immer preisgünstiger und einfacher zu bekommen. Sisyphus war dann meine erste CNC-Fräse, die kein Mittel mehr dazu war, eine Skulptur zu erstellen, sondern die Skulptur selbst war.“

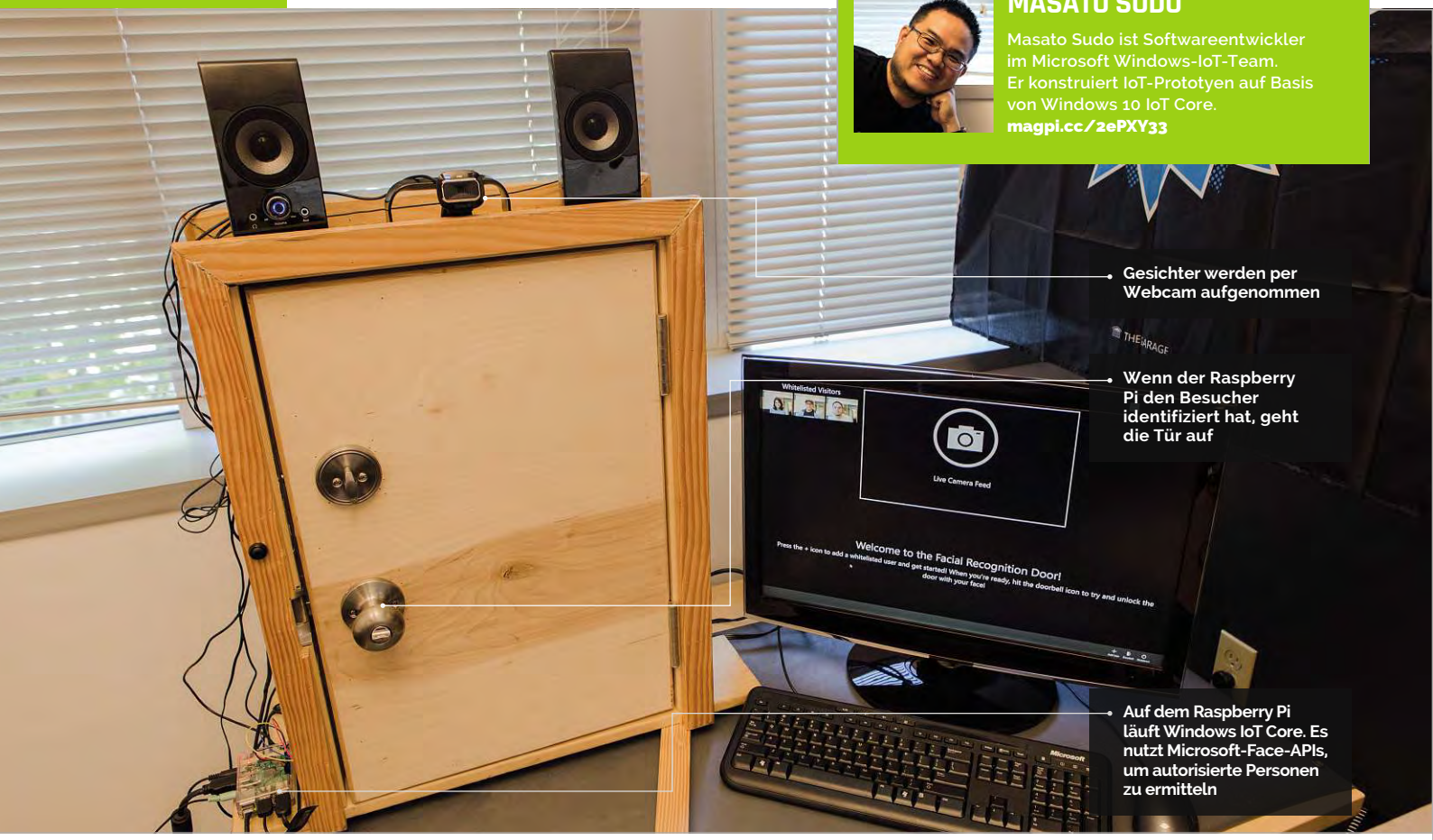
Oben Die Beleuchtung des Tisches lässt sich regulieren, ebenso wie die Geschwindigkeit der Metallkugel

Zwanzig Jahre hat Bruce nun bereits an Sisyphus getüftelt – dennoch war er anfangs voller Zweifel, ob der Marktstart glücken würde. Er hätte sich darüber keine Sorgen machen müssen. Ziel der Kickstarter-Kampagne war es, 50.000 US-Dollar einzusammeln. Am Ende kamen fast zwei Millionen zusammen.

„Kickstarter ist eine ideale Plattform, um zu zeigen, was du vorhast und um herauszufinden, wie viele Leute sich dafür interessieren, und zwar nicht nur mit warmen Worten, sondern mit ihrem schwerverdienten Geld“, resümiert Bruce.

MASATO SUDO

Masato Sudo ist Softwareentwickler im Microsoft Windows-IoT-Team. Er konstruiert IoT-Prototyen auf Basis von Windows 10 IoT Core.
magpi.cc/2ePXY33



WINDOWS IOT: EINE TÜR MIT GESICHTSERKENNUNG

Infos

- Die Tür ist während Microsofts One-Week-Hackathon für Mitarbeiter entstanden
- Das Projekt nutzt die Face-API der Microsoft Cognitive Services
- Um die Tür zu öffnen, wird ein Leviton 79A00 eingesetzt
- Außerdem kommt eine MS Lifecam HD3000 zum Einsatz

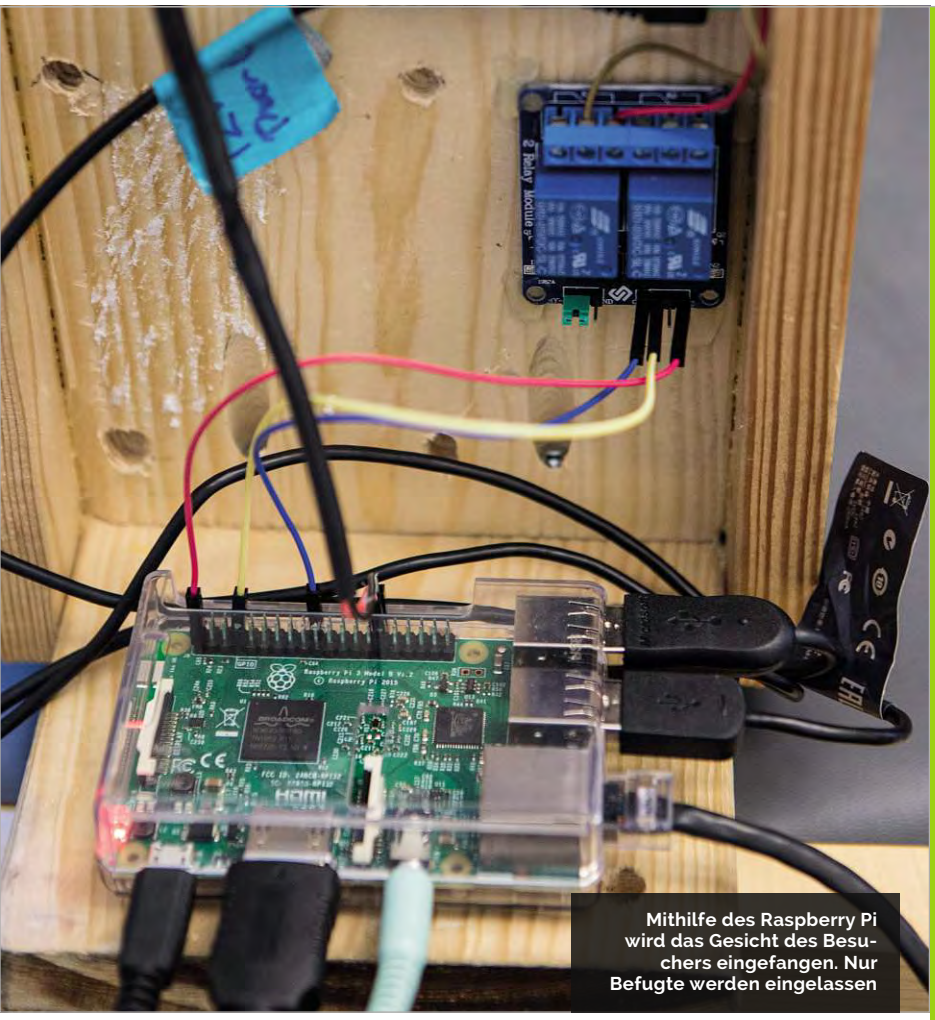
Diese RasPi-gesteuerte Tür lässt nur Besucher ein, die sie identifizieren kann. Alle anderen bleiben außen vor

Microsoft hat es sich zum Anliegen gemacht, mithilfe von Windows 10 IoT Core möglichst viel aus dem Pi herauszuholen. Zwar ist Windows 10 IoT Core vielleicht nicht gerade die Windows-10-Home-Version, die sich manche User für ihren Raspberry Pi wünschen. Dennoch bringt es einige wirkliche mächtige Dienste mit und ermöglicht es Makern, spannende Projekte auf die Beine zu stellen.

„Die Tür mit Gesichtserkennung, die auf Windows 10 IoT Core basiert, ist ein Beispiel dafür, was für fesselnde Dinge entstehen können, wenn Windows IoT Core mit Microsofts kognitiven Services kombiniert wird“, so Masato Sudo, Softwareentwickler im Windows-IoT-Team von Microsoft. Er erläutert: „Bislang handelt es sich um einen Prototypen, der Besucher identifizieren kann.“ Wenn die Tür den Besucher „erkennt“, spricht sie ihn mit seinem Namen an, und die Tür öffnet sich. „Wenn sie den Besucher nicht erkennt, teilt sie ihm das mit – und die Tür bleibt verschlossen wie eine Auster.“

Gesichtserkennung ist eine beträchtliche Herausforderung, aber die Anforderungen an die Hardware sind erstaunlich gering. Der Raspberry Pi ist mit einer Standard-USB-Webcam verbunden, um die Gesichter von potenziellen Besuchern einzufangen. Außerdem wird ein Lautsprecherset eingesetzt, um dem Besucher akustisch Feedback geben zu können. Damit die Tür sich öffnen kann, wird ein GPIO-Pin mit einem Schaltrelais verbunden. Schickt der Pi ein entsprechendes Signal an die GPIO, gibt das Schaltrelais die Tür frei. Ein weiterer GPIO-Pin wird mit der Türglocke verbunden. Wenn der Besucher klingelt, wird er fotografiert, und die Gesichtserkennung startet.

„Windows 10 IoT Core ist das Gehirn dieses Projekts. Für die Gesichtserkennung nimmt es eine API der Microsoft Cognitive Services in Anspruch, die in der Azure



Mithilfe des Raspberry Pi wird das Gesicht des Besuchers eingefangen. Nur Befugte werden eingelassen

Cloud läuft. Microsoft Cognitive Services (**magpi.cc/2eDbDM6**) befassen sich mit Machine Learning und künstlicher Intelligenz in den Bereichen Sprache, Sehen, Wissen und Recherche“, erläutert Masato. Für das Tür-Projekt setzte er die Gesichtserkennungs-API ein (**magpi.cc/2eDb3xV**).

> Wenn die Tür den Besucher nicht erkennt, bleibt sie verschlossen wie eine Auster

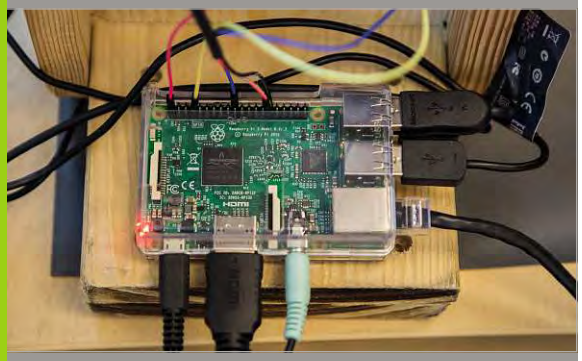
„Wir erstellen eine Liste mit unseren Freunden und der Familie, und versehen sie mit den entsprechenden Fotos.“ Diese Liste wird in der Microsoft Azure Cloud hinterlegt. Nachdem ein Bild des Besuchers geschossen wurde, wird erst einmal festgestellt, ob es sich bei der Aufnahme wirklich um ein menschliches Gesicht handelt, bevor sie an die Gesichtserkennung geht. Ist das der Fall, wird das Foto an die Microsoft Azure Cloud geschickt und die Person gegebenenfalls identifiziert. „Die Aufnahme wird mit der Liste abgeglichen, die wir erstellt haben“, so Masato, „die Trefferquote wird dann anhand einer Skala von 0-100 Prozent ausgegeben.“

Er fährt fort: „Wir haben den Code in C# geschrieben und Visual Studio 2015 eingesetzt, um das Projekt auf dem Pi zum Laufen zu bringen. Den Code haben wir für die Open-Source-Community freigegeben, sodass Entwickler zur Verbesserung der Sicherheit beitragen können.“

SO FUNKTIONIERT DIE ERKENNUNG

>SCHRITT-01
Draußen vor der Tür

Der Besucher drückt die Türglocke, eine einfache Webcam nimmt sein Gesicht auf. Über die Lautsprecher wird der Besucher akustisch begrüßt.



>SCHRITT-02
Gesichtserkennung

Auf dem Raspberry Pi läuft Windows 10 IoT Core. Das Programm schickt das Foto an die Microsoft Cognitive Services, wo es mit der Liste der Personen abgeglichen wird, die Zutritt zum Haus haben.

>SCHRITT-03
Besucher einlassen

Führt der Abgleich zu einem Treffer, wird die Tür mittels Schaltrelais geöffnet, und der Besucher wird mithilfe der Lautsprecher namentlich willkommen geheißen.

MATT LONG & MARK NICHOLS
Matt Long und Mark Nichols (im Bild) sind im Team von Microsoft Cloud Solution und arbeiten an der Pegasus-Mission mit. **pegasusmission.com**

PEGASUS UND DER NORTH AMERICAN EAGLE

Infos

- Das Fahrzeug ist 17 m lang
- Es hat 42.500 PS
- Die Räder sind aus Aluminium
- Diese Räder überstehen Geschwindigkeiten von über 1.290 km/h
- Das Gefährt schluckt während des Rennens bis zu 340 Liter Benzin pro Minute

Im ehemaligen Düsenjet North American Eagle sammelt nun ein Raspberry Pi Daten und verschickt diese in Echtzeit

Im Cockpit eines der bemerkenswertesten Fahrzeuge der Welt sitzt ein Raspberry Pi: Mit dem „North American Eagle“, einem ausrangierten Düsenjet, versucht ein amerikanisches Team, einen neuen Geschwindigkeitsrekord für Landfahrzeuge aufzustellen. Dieser liegt derzeit bei 763 Meilen beziehungsweise 1.227,9 Kilometern pro Stunde. Aufgestellt wurde der Rekord 1997 vom britischen Royal-Air-Force-Piloten Andy Green in seinem düsengetriebenden Fahrzeug „Thrust SSC“.

Rechts Ein Raspberry Pi erhebt Sensordaten aus dem Cockpit

„Wir wollen den Weltrekord brechen“, meint Brandyn Bayes, Mitglied im Team des North American Eagle. „Uns stand nur ein abgewrackter Düsenjet zur Verfügung, der noch aus den Zeiten des Kalten Krieg stammte – daraus haben wir einen der ausgefeiltesten Rennwagen aller Zeiten entwickelt.“

Der North American Eagle war ursprünglich ein Lockheed F-104 Starfighter. „Er wurde als Jagdflugzeug eingesetzt“, erklärt uns Brandyn. Diese Maschinen konnten Forschungsflugzeuge begleiten und technische Messungen durchführen. Die F-104, die in ein High-Speed-Landfahrzeug umgewandelt wurde, stammt aus dem Jahr 1957. 1998 war sie eigentlich ein Haufen Schrott. Durch das Projekt „North American Eagle“ wurde sie zu neuem Leben erweckt – und entwickelte sich wieder zu einem der schnellsten Fahrzeuge der Welt.

Während der North American Eagle dem Geschwindigkeitsrekord zu Lande hinterherjagd, ist die Mission Pegasus eigentlich eher Richtung Himmel orientiert. „Die Mission Pegasus nutzt Höhenballons, um Nutzlasten zu transportieren“, erklärt uns Matt Long, Mitarbeiter im Microsoft Cloud-Solution-Team. „Der Ballon ist vollgepackt mit Sensoren, um in Echtzeit Telemetriedaten zu übertragen und die Flugbewegungen zu überwachen.“

Genau diese Erfahrung in Echtzeitkommunikation hat Matt und Mark Nichols (ebenfalls im Microsoft Cloud-Solution-Team) mit dem Projekt North American Eagle zusammengeführt. „Wir boten dem Team des North American Eagle die Zusammenarbeit an“, so Mark. Der Pegasus-Ballon „liefert den Nutzern eine spannende Erfahrung, weil sie die Flüge live mitverfolgen können“, erläutert Mark. „Ziel war es also auch in diesem Fall, Zuschauern auf der ganzen Welt in Echtzeit zu übermitteln, was während der Fahrt des North American Eagle geschieht. Für dieses Projekt entwickelte ich eine maßgeschneiderte Lösung – auf Basis eines Raspberry Pi, auf dem Windows 10 IoT Core lief. Dieses Gerät wurde im Cockpit des Gefährts platziert. Die Software wurde in C# geschrieben.

„Microsoft unterstützt das Projekt North American Eagle bereits eine ganze Weile“, so Mark. „Freiwillige haben sich zusammengetan, um die Hardware, die Software, die Cloudanbindung und die Mobil- und Webapps aufzusetzen.“

> „Hauptaufgabe des Raspberry Pi war es, das Fahrzeug während der Testfahrt zu analysieren“

Pi übermittelt Daten

Die Hauptaufgabe des Raspberry Pi war es, das Fahrzeug während seiner Testfahrt zu analysieren. „Er hat in Echtzeit GPS-Daten, Informationen zur Beschleunigung und zur Geräuschentwicklung erhoben und an die Microsoft Cloud geschickt. Das Pegasus-Team konnte die Vorzüge der Echtzeitübermittlung von Daten einbringen, um das Team des North American Eagle seinem ehrgeizigen Ziel näherzubringen“, sagt Mark.

HIGH-SPEED-TELEMETRIE

>SCHRITT-01

Das Gerät

Ein Raspberry Pi bildet das Herzstück des Telemetriegeräts. Er ist gespickt mit Sensoren und wird im Cockpit des North American Eagle angebracht.

>SCHRITT-02

Azure-Cloud

Während des Rennens verbindet sich der Raspberry Pi mit der Microsoft Azure Cloud. Er läuft unter Windows IoT Core und nutzt Software, die in C# geschrieben ist.

>SCHRITT-03

Während des Rennens

Fans können sich während des Rennens einklinken und dem Team Mitteilungen schicken, die während des Rennens angezeigt werden. Telemetriedaten werden in Echtzeit ans Begleitteam übertragen.

Der unterhaltsamere Part war die Echtzeit-Interaktion mit den Beobachtern während der Testfahrt. „Leute, die unsere Mobil-Apps nutzen, konnten uns Unterstützungsbotschaften schicken. Sie landeten direkt im Fahrzeug, wo sie auf einem LC-Screen angezeigt wurden“, erzählt Mark.

Der Bildschirm war hinter dem Fahrer angebracht. Hier installierte Mark zusätzlich eine GoPro-Kamera, um die Botschaften aufzuzeichnen. „Der Fahrer konnte die Mitteilungen nicht sehen, sie hätten ihn zu sehr abgelenkt. Aber die Kamera nahm das Display und das Cockpit auf.“

Natürlich musste das Team die Userkommentare filtern, um Unangebrachtes zu entfernen. „Das System ist dafür ausgerichtet, Userkommentare mittels eines Drittanbieterdienstes namens Web Purify zu überprüfen“, so Mark. „Erst nach dieser Kontrolle wird die Botschaft auf das Display im Cockpit ausgespielt.“

Matt fährt fort: „Die Kamera hatte einen zusätzlichen Vorteil: Sie gewährte uns den Blick aus dem Cockpit heraus nach vorn.“ Diese Videoaufzeichnung wurde herangezogen, um mit Jessi Combs, einem der Testfahrer, die erste Probefahrt zu analysieren.

Ab in die Azure-Cloud

Matt ergänzt: „Wir haben zusätzlich zwei Drohnen zur Videoaufzeichnung angeschafft. So konnten wir die Testfahrten aus vielen Perspektiven beobachten und gleichzeitig dieses Event, das an einem sehr entlegenen Ort stattfand, in all seinen Facetten aufzeichnen.“

Die Technik, die all dies ermöglicht, ist Azure. „Azure liefert uns weltweite Kommunikation in Echtzeit“, erklärt er. „Wir können Telemetriedaten erheben, oder auch Mitteilungen an Nutzer ausgeben, und die Daten zeitgleich sogar analysieren.“

Matt erzählt, dass die an Bord des North American Eagle erhobe-

nen Daten bei Azure hochgeladen und betrachtet werden können. „Ein Team aus etwa 20 Microsoft-Volunteers hat Xamarin-basierte Apps für iOS, Android und Windows Mobile entwickelt."

„Vom ersten Tag an gab es Leute, die das Projekt für leicht verrückt hielten", sagt Brandyn vom North-American-Eagle-Team. „Vielleicht ist das sogar eine Untertreibung. Aber genau das Abenteuerliche daran hat uns angetrieben – das Gefühl, etwas zu tun, was kaum jemand wagen würde."

Fans sind live dabei

Die Microsoft-Research-Abteilung hat es dem Team ermöglicht, den Fans der North American Eagle Echtzeitdaten zur Verfügung zu stellen, während das Gefährt durch eine unwirtliche Wüste raste.

„Wir haben Mitteilungen von Fans aus Australien, Neuseeland, Großbritannien, Frankreich, Norwegen und den USA erhalten. Einer der Nutzer, die die erste Testfahrt verfolgten, schickte uns eine Einschätzung der erreichten Höchstgeschwindigkeit – während das Rennen noch lief."

„Leider haben wir den Rekord nicht gebrochen", sagt Brandyn, „aber das Projekt hat uns ins Gedächtnis gerufen, warum wir Rennen lieben: Wenn alles gut läuft, sorgt das für einen richtigen Adrenalinkick."

„Das Team des North American Eagle, Ed Shadle und Jessi Combs, ist bis ans Limit gegangen, um einen neuen Geschwindigkeitsrekord aufzustellen", ergänzt Mark. „Und das hat uns alle mit großem Stolz erfüllt."

Oben Der North American Eagle nutzt seinen Düsenantrieb, um bahnbrechende Geschwindigkeiten zu erreichen

CLODAGH O'MAHONY

Nach ihrem Master-Abschluss in Creative and Innovative Design Technology arbeitet Clodagh freiberuflich. Sie ist außerdem sehr gut im Rudern und im Bogenschießen.

clodaghomahony.com

QBEE INNOVATIVE SOCIAL-MEDIA-PLATTFORM

Clodagh O'Mahonys Kleid speichert Berührungen und Sprachdaten und vergibt Punkte für soziale Interaktion

Infos

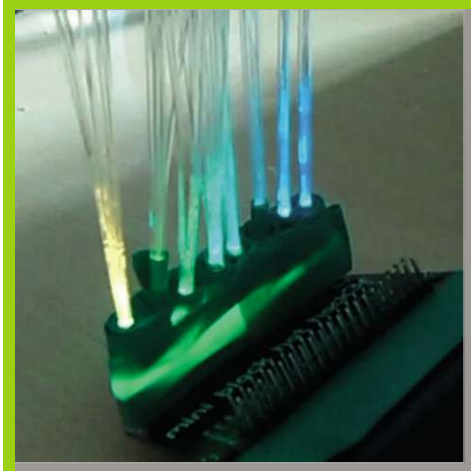
- Das Kleid wurde umgearbeitet, um den Tragekomfort und die Funktionalität zu verbessern
- Der Pi zeichnet Berührungen und auch Keywords auf
- Die Daten werden in eine SQL-Datenbank hochgeladen
- Mit leitendem Faden wurde das Kleid in Sektoren aufgeteilt

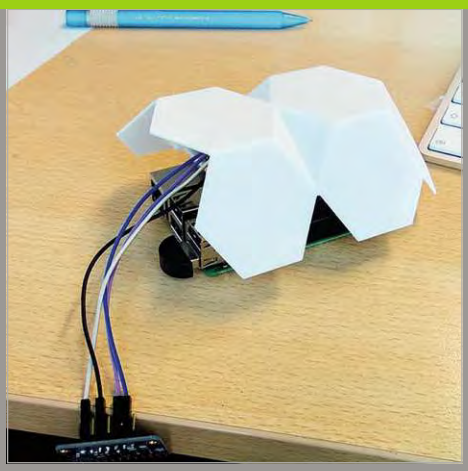
Clodagh O'Mahony beschreibt sich selbst als „multidisziplinäre Designerin mit Erfahrung in Produkt-, Graphik- und UX/UI-Design, sowie Illustration and Mediaproduktion." Nach ihrem Bachelor-Studium in Product Design and Technology an der University of Limerick beendete Clodagh ihr Studium 2016 mit einem Master in Interactive Media.

Für ihr Abschlussprojekt kreierte Clodagh ein spezielles Kleid und erstellte begleitend dazu eine Webseite, um die Social-Media-Interaktion abzubilden und zu kommentieren. Der Gedanke dahinter: Es wird immer schwieriger, sich auf Plattformen wie Facebook und Twitter bedeckt zu halten – durch die schiere Masse an persönlichen Informationen, die wir in unserer Timeline preisgeben. Früher konnte man in rein textbasierter Kommunikation auf Blogs und in Chat Rooms einfach ein Alter Ego aufbauen, das sich mit der tatsächlichen Person nicht in Verbindung bringen ließ. Heutzutage verwenden Social-Media-Nutzer jedoch vermehrt persönliche Fotos und Videos. Deswegen entwickelte Clodagh „eine tragbare, vernetzte Plattform, die eine neue, reinere Art von Social Media einführen soll. Da sie mit quantitativen Daten arbeitet, müssen Nutzer unverhältnismäßig große Anstrengungen unternehmen, um ihr Leben verfälscht darzustellen. Dadurch sind die gesammelten Daten zuverlässiger als die anderer Plattformen."

Clodagh erfand dafür die fiktive Firma „QBee". Der Name ist eine Abkürzung von Queen Bee (Bienenkönigin). Das damit verbundene Honigwaben-Design spielt für das Aussehen des Kleides und der dazugehörigen Webseite eine große Rolle. Das Unternehmen würde, wenn es real existierte, eine Produktlinie aus tragbarer Technik verkaufen. Diese wäre, wie das Kleid aus der Abschlussarbeit, dazu in der Lage, Daten sozialer Interak-

ELEKTRONIK GESCHICKT EINGEFÄDELT





>SCHRITT-01
Beleuchtung
Die Lichtwellenleiter sind in Bündeln an die RGB-LEDs des Blinkt! angeschlossen. So können die einzelnen Farben einfacher über das ganze Kleid verteilt werden.

>SCHRITT-02
Gehäuse
Die im 3D-Drucker erstellten Gehäuse bilden die hexagonale Form einer Honigwabe nach. Das Thema wiederholt sich im Kleid und auf der Webseite.

>SCHRITT-03
Anprobe
Für einen optimalen Tragekomfort wurden der Raspberry Pi und die anderen elektronischen Bauteile in diesen Gehäusen auf dem Rücken befestigt.

tionen aufzunehmen und im QBee-Account des Käufers hochzuladen.

Ziel des Projekts war es, physische Interaktion zwischen dem Träger und den Mitmenschen aufzuzeichnen. Eine Berührung an der Taille etwa ist von der Berührung am Rücken unterscheidbar und wird zusätzlich durch farbiges Aufleuchten angezeigt. Außerdem lauscht ein Mikrofon mit und achtet auf Keywörter, die in einer vordefinierten Liste positiv oder negativ bewertet wurden.

> „Eine tragbare, vernetzte Plattform, die eine neue Art von Social Media einführen soll"

Clodagh verwendete dafür ein Adafruit 12-Key Capacitive Touch Sensor Breakout Board, Pimoroni Blinkt!, Lichtwellenleiter und einen Raspberry Pi. Sie sind in speziellen hexagonalen Gehäusen untergebracht, die mit einem 3D-Drucker erstellt wurden. Das Blinkt! und die Lichtwellenleiter sollten ursprünglich die Datenerfassung bunt hervorheben: Bei einer Berührung an der Taille beginnt das Kleid lila zu leuchten, eine Berührung an der Hüfte lässt es grün aufleuchten.

Clodagh hat das Kleid mehrmals umgearbeitet, um Faktoren hinsichtlich Kosten, Komfort und Usability zu berücksichtigen, die ihr erst während der Arbeit auffielen. Sie hat alle Schritte in ihrem Instagram-Account dokumentiert (**magpi.cc/2eJgHuZ**). Im ursprünglichen Entwurf wurde die Technik zwar wie gewünscht umgesetzt, aber das Kleid war zu unbequem. Am Ende verzichtete Clodagh deshalb auf manche technischen Details zu Gunsten des Tragekomforts. So hat sie den Plan mit den unterschiedlichen Zonen aufgegeben und das Blinkt! durchläuft nun bei Berührung des Kleides, egal an welcher Stelle, eine Regenbogensequenz, was zumindest die Idee von Datenerfassung in Kombination mit Berührungssensoren demonstriert.

Ausstellung des Abschlussprojekts

Oben Clodagh experimentierte mit verschiedenen Gehäuse-Designs, bevor sie schließlich die optimale Lösung fand

WENDELL KAPUSTIAK

Der Investmentbanker im Ruhestand lebt in Venice, Florida. Er arbeitete schon immer gern mit Holz. Seinen Raspberry Pi nutzt er, um die Daten seiner Wetterstation zu verarbeiten und seinen Stromverbrauch zu kontrollieren.

SELBSTSPIELENDE PFEIFENORGEL

Dieses handgearbeitete Musikinstrument spielt jede beliebige MIDI-Datei ab

Infos

- Anfangs wollte Wendell einen MIDI-Weinglas-Player basteln
- Er brauchte etwa drei Jahre, um die Orgel zu konstruieren
- Das Gebläse erstand er für 80 US-Dollar auf Ebay
- Die Pfeifen sind zwischen 23 und 100 cm lang
- Derzeit baut er die Orgel noch einmal neu – in massiver Eiche

Wendell Kapustiak hat durchaus Erfahrung mit Holzarbeiten – aber um diese selbstspielende Orgel zu bauen (**magpi.cc/2fMacLy**), musste er wirklich alles geben. „Das größte Problem war, dass ich keine Ahnung hatte, wie eine Pfeifenorgel eigentlich funktioniert“, gibt Wendell zu. Letzendlich griff er für die Mechanik auf ein Projekt von Matthias Wandel zurück (**woodgears.ca**). Der schwierigste Part war, die 42 Holzpfeifen herzustellen, da jede von ihnen andere Abmessungen hat, um die gewünschte Tonhöhe zu erreichen. Insgesamt beträgt der Tonumfang dreieinhalb Oktaven. Für die Umsetzung griff Wendell auf Tutorials zurück, die Raphi Giangiulio auf YouTube bereitgestellt hat (**youtu.be/-mibK_Dp-ZY**).

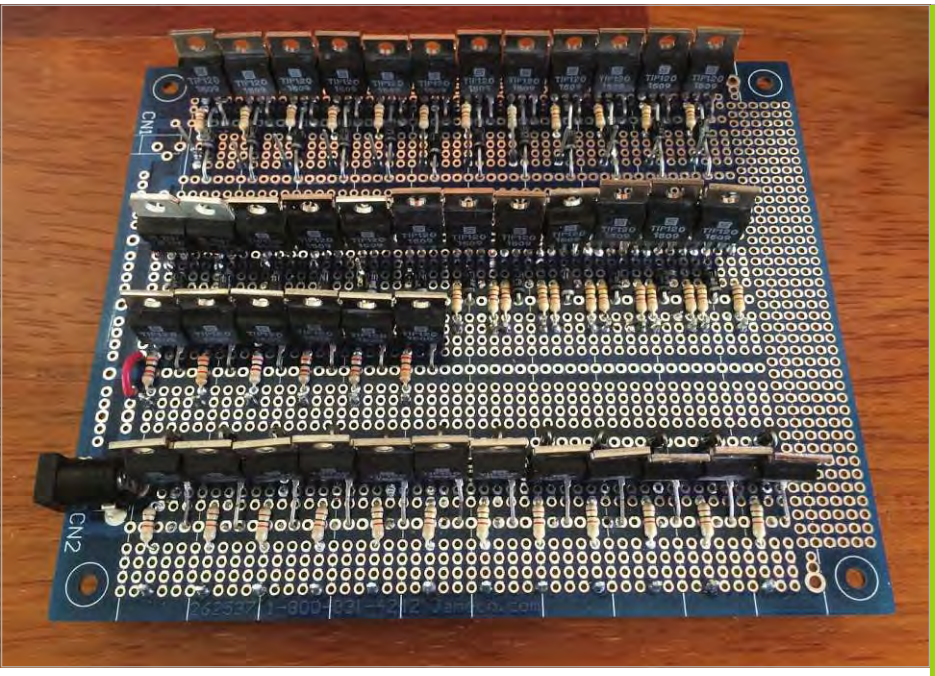
Die Pfeifen werden über PVC-Rohre mit einer Windlade verbunden. Diese wird von einem Kooltronic-KBR125-Gebläse angetrieben, wie sie in Rechenzentren genutzt werden. „Anfangs hatte ich es mit einem kleinen Industriestaubsauger versucht“, erzählt Wendell. „Damit gab es aber zwei Probleme. Erstens war er furchtbar laut. Zweitens lief er schnell heiß. Als ich ihn in eine Box legte, damit er weniger Krach macht, wurde er so heiß, dass er mir abrauchte.“

Um sie zum Spielen zu bringen hat jede Holzpfeife ein Ventil, das

Jede der handgearbeiteten Holzpfeifen bringt einen anderen Ton hervor

Wenn sie ausgelöst werden, öffnen die Magnetspulen die entsprechenden Ventile

Die Magnetspulen sind indirekt mit einem Arduino Due verbunden, der Daten von einem Raspberry Pi bezieht



per Magnetspule geöffnet und geschlossen wird. Diese wird von einem Arduino Due mittels Hochleistungs-Treiberplatine angetrieben. „Gehirn“ des Ganzen ist ein Raspberry Pi, der mehrere Funktionen hat. Einerseits bietet er ein grafisches User-Interface, um Musikstücke auswählen zu können.

Oben Ein selbstgebasteltes Treiber-Board sorgt für die nötige Spannung, um die Magnetspulen auszulösen

> Der Raspberry Pi beachtet Tempiwechsel und andere musikalische Feinheiten

Außerdem wandelt er die MIDI-Files in spielbare Befehle um, inklusive der Musikrichtung. „Der Pi berücksichtigt Tempiwechsel und andere Feinheiten der Notation“, erläutert Wendell. „Auf diese Weise erledigt er einen Großteil der Arbeit – so weit das durch Berechnung möglich ist.“ Mithilfe eines weiteren Python-Programms werden die Daten dann über USB an den Arduino geschickt.

Während dieser Teil des Projekts reibungslos vonstatten ging, brachte die eigentliche Konstruktion einige Probleme mit sich. Eines davon war der Mechanismus, der die durchströmende Luft reguliert. „Ich habe es mit verschiedenen Bauweisen probiert. Der Mechanismus sollte möglichst einfach und zuverlässig sein und für eine stabile Luftzufuhr sorgen.“ Glücklicherweise bekam Wendell eines Tages Besuch von seinem Freund Jim. Dieser, ein ehemaliger Ingenieur für Klimatechnik, schlug vor, es mit einer ganz anderen Methode zu versuchen. „Ich sollte eine Tür einsetzen, die durch ein Gewicht gesteuert wird – so wie es bei Klimasystemen gemacht wird. In nur 20 Minuten hatte ich einen Prototyen aufgebaut: Es funktionierte perfekt.“

Eine knifflige Angelegenheit war auch die Anordnung der Pfeifen. Sie dürfen nicht zu nahe beieinanderstehen, um sich nicht gegenseitig zu stören. Außerdem änderte Wendell am Schluss noch einmal die gesamte Anordnung und brachte die Magnetspulen vorn an. So können die Zuhörer sehen, wie die Ventile bewegt werden. Zustätzlich hat er noch einige LEDs angebracht. Das macht es dem Publikum noch einfacher, zu erkennen, was vor sich geht.

SO ENTSTEHT DIE ORGEL

>SCHRITT-01
Die Holzpfeifen

Jede der 42 Pfeifen bringt einen anderen Ton hervor. Längere Pfeifen erzeugen tiefere Töne. Manche ganz tiefe Töne kommen aber von kürzeren Pfeifen, bei denen die Tonhöhe mechanisch herabgesetzt wird.

>SCHRITT-02
Die Ventile

Die Pfeifenventile werden mittels Magnetspulen geöffnet und geschlossen. O-Ring-Dichtungen sorgen dafür, dass die Ventile nicht festhängen und dass keine klackenden Geräusche entstehen.

>SCHRITT-03
Die Windlade

Die Windlade verfügt über eine Öffnung, die mit einem Gewicht versehen ist. Spielen viele Pfeifen gleichzeitig, schließt die Öffnung automatisch, um den Luftdruck zu erhöhen.

INITIAL STATE

PIOT CHALLENGE GEWINNER

BRIAN BRANDAW

ist schon seit 25 Jahren in der IT tätig und hat jede Menge Erfahrung mit Linux-Systemen. Zu Hause experimentiert er gern mit seinem RasPi

WASSERSTAND AUTOMATISIERT ABLESEN

Der Gewinner des Initial State PiOT Challenge kann mithilfe eines Raspberry Pi den Füllstand seiner Regentonnen berechnen

Infos

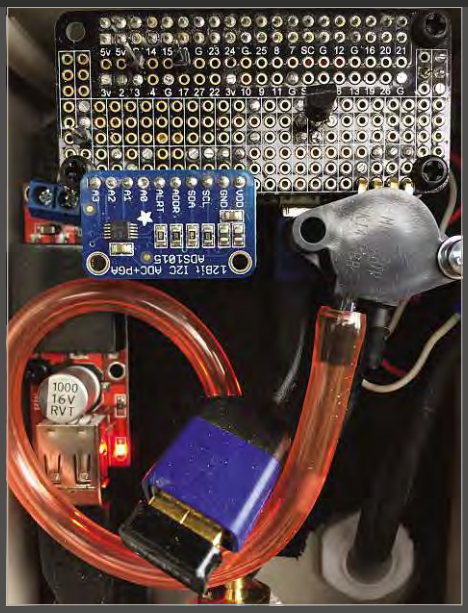
- Das Projekt hat etwa vier Monate in Anspruch genommen
- Brian schätzt, dass er es nun in etwa vier Stunden nachbauen könnte
- Das ProtoZero-Board kann man direkt mit der GPIO verbinden
- Es lässt sich drahtlos steuern
- Brian hat noch weitere Pis, die Teile seines Grundstücks überwachen

Wir haben in der MagPi bereits etliche Projekte vorgestellt, bei denen der Raspberry Pi für Automationsaufgaben eingesetzt wird. Das Projekt von Brian Brandaw aus Texas ist aber etwas ganz Besonderes: Es misst den Füllstand in einer Reihe von Wassertonnen.

„Wir wohnen in einer ländlichen Gegend und beziehen unser Wasser aus einem privaten Brunnen“, erklärt Brian. „Die Wasserqualität ist aber ziemlich schlecht und der Brunnen ist nicht sehr zuverlässig. Daher haben wir zwei große Regentonnen aufgestellt, die das Wasser vom Dach unserer Scheune und aus allen Regenrinnen sammeln. Wir haben sie mit zwei handelsüblichen Schwimmerventilen versehen, um damit die Pferdetränke mit Wasser zu versorgen. Das hat sehr gut funktioniert – bis eines Tages die Regentonnen leer waren. Was fehlte, war eine möglichst bequeme Methode, um den Wasserstand in den Tonnen abzulesen.“ Üblicherweise wird das einfach visuell erledigt, das heißt Sie müssen regelmäßig nach draußen gehen und die Wasserstandanzeige überprüfen. Brian fährt fort: „Ich wollte den Füllstand der Wassertonnen aber jederzeit im Blick haben und überdies den Wasserverbrauch beobachten.“

DEN WASSERSTAND KONTROLLIEREN

>SCHRITT-01
Der Drucksensor
Der Sensor auf dem Boden der Tonne steht immer unter Druck. Dieser verändert sich jedoch, je nachdem, wie viel Wasser im Behälter ist



>SCHRITT-02
Helferlein für den Pi
Damit der Pi Zero die Daten auslesen kann, wird zusätzlich ein Analog-/Digitalwandler benötigt. Die GPIO selbst ist dazu nicht in der Lage.

>SCHRITT-03
Pi-Berechnungen
Der Pi Zero nutzt die Fließbewegung und die Messungen in der Tonne, um den Füllstand zu bestimmen. Dieser wird auf einem Chart angezeigt - so haben Sie ihn immer im Blick.

Natürlich sollte das Ganze via Computer funktionieren – wir leben ja schließlich im 21. Jahrhundert. Brians Verfahren ist ziemlich einzigartig: Statt an bestimmten Messpunkten den Wasserstand abzulesen, misst er den Wasserdruck am Boden der Tonne. Berücksichtigt man auch den Luftdruck und die Größe des Behälters, kann man damit den Füllstand errechnen.

> „Ich will den Wasserstand in meinen Regentonnen immer unter Kontrolle behalten"

„Ich habe einen Pi Zero eingesetzt, um mittels Analog-Digitalwandler den Drucksensor auszulesen, den Füllstand zu berechnen und die Daten an InitialState zu übermitteln. Dort werden sie gesammelt und ausgespielt", erläutert Brian. Während seiner beruflichen Laufbahn hat er Erfahrungen mit Linux gesammelt. Das hat es ihm erleichtert, dieses Projekt aufzusetzen. Er konnte auf eine ähnliche Arbeit zurückgreifen, die er vor einigen Jahren mithilfe eines Arduino umgesetzt hat. „Das Ganze war ein bisschen tricky, aber nicht deshalb, weil das Projekt so komplex ist. Die Box, die ich ausgewählt habe, ist recht klein, sodass ich einige Zeit brauchte, um alles hineinzuquetschen. Die größte Herausforderung war der limitierte Platz auf dem ProtoZero-Board. Ich bin nicht sehr gut im Löten, daher hat das seine Zeit gebraucht."

Brian freut sich, dass das Ganze so gut funktioniert. Nur einmal war eine kleinere Reparatur fällig, weil er ein Verbindungsstück beschädigt hatte. Insgesamt hat dieses Projekt seine Wasserversorgung erheblich verbessert.

Oben Die kleine graue Box mit den Geräten wird mit der Regentonne verbunden

RaspCade-Schablone auf Heft-DVD

WESLEY ARCHER

Der Raspberry-Pi-Fan betreibt die Website Raspberry Coulis und schreibt Anleitungen für Pi Supply and Cyntech.
raspberrycoulis.co.uk
@RaspberryCoulis

Sie brauchen

- RaspCade-Schablone **(auf DVD)**
- Zugang zu einem Lasercutter
- Holzleim
- 2 × 3-Zoll, 4Ω, 3W-Lautsprecher **(ModMyPi.com)**
- 8× versenkbare M4-Muttern und -Schrauben
- Ein Bohrer mit einem Senkbohr-Aufsatz
- Mehrere 12,7-mm-Abstandshalter
- kleine Magnet-verschlüsse

DER EIGENE RASPCADE: DAS GEHÄUSE MONTIEREN

Im vierten Teil unserer Bauanleitung zeigen wir Ihnen, wie Sie aus Ihrem vorgefertigen Gehäuse einen RaspCade zusammensetzen

Im ersten Teil unseres Workshops haben wir unser RaspCade-Gehäuse bereits kurz vorgestellt. In dieser Ausgabe erläutern wir, wie der Zusammenbau des Gehäuses genau funktioniert. Außerdem werden wir unser RaspCade noch mit Lautsprechern versehen und das „Innenleben" vormontieren – in Vorbereitung auf das große Finale in der nächsten Ausgabe, wenn wir unseren Arcade-Automaten endlich zum Laufen bringen.

>Schritt-01

Die Bauanleitung

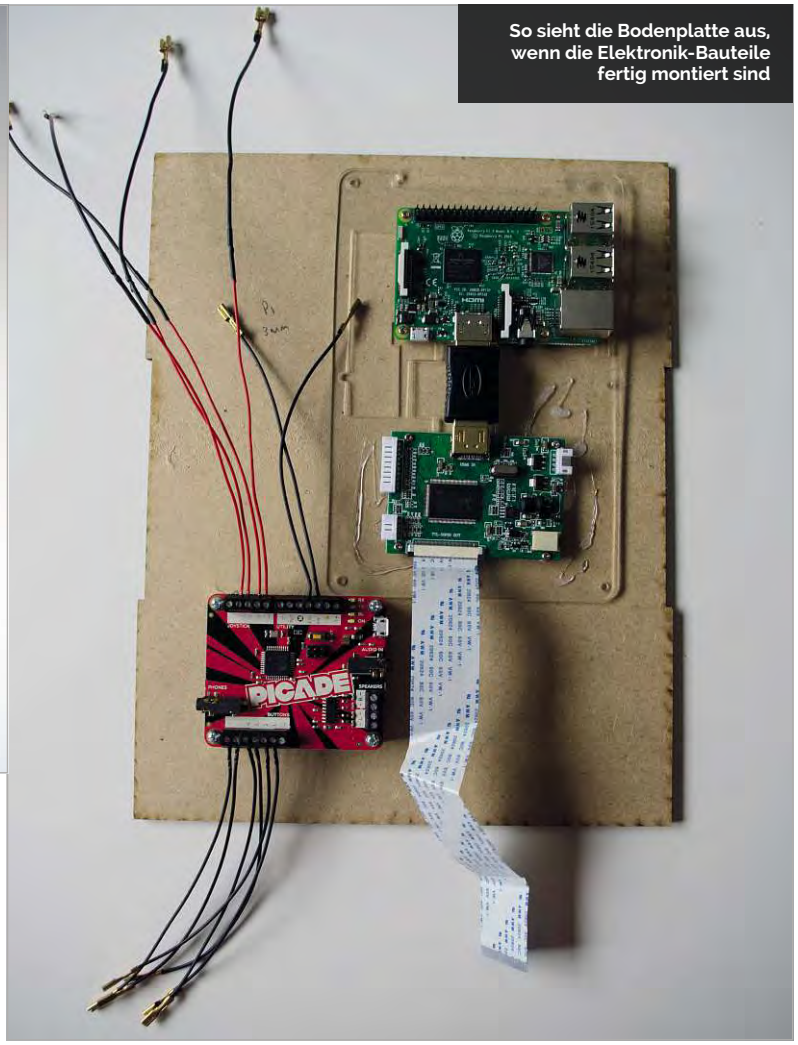
Wir stellen Ihnen die Schablone auf unserer Heft-DVD bereit, alternativ können Sie sie auch unter **magpi.cc/21jHT61** herunterladen. Die Schablone liegt im EPS-Format vor und ist damit die perfekte Vorlage für Onlinedienste, die solche Modelle per Laser zurechtschneiden. Sie können die Pläne also einfach herunterladen und sie bei einem Onlinedienst Ihrer Wahl wieder hochladen.

>Schritt-02

Bohrer bereitlegen

Es kann sein, dass Sie das Gehäuse für Ihre Zwecke anpassen und eigene Löcher für die Montage bohren müssen. Nachdem wir den Bildschirm des Qubit-Case verwendet haben, war die Montage von LCD-Driver-Board und Pi in unserem Fall unkompliziert. Aber es stellt auch keine große Hürde dar, die Löcher für die Montage und für die Abstandshalter selber vorzubohren. In unserem Testaufbau haben wir einen kleinen Bohrer benutzt,

Die Lautsprecher werden seitlich ans Gehäuse montiert



So sieht die Bodenplatte aus, wenn die Elektronik-Bauteile fertig montiert sind

um einige Löcher für die Lautsprecher vorzubohren und darüber hinaus Befestigungspunkte, an denen wir Abstandshalter für die Picade PCB angebracht haben.

>Schritt-03

Die Lautsprecher montieren

Die Lautsprecher anzubringen, ist sehr einfach. Die Modelle, die wir im Testaufbau verwenden, sind jeweils mit vier Befestigungslöchern versehen, mit denen sie sich am Gehäuse montieren lassen. Wir bringen also unsere Lautsprecher in die richtige Position, markieren per Bleistift, wo die Bohrlöcher hingehören, und bohren dann die Löcher. Mit dem Senkbohrer schrägen wir die Löcher an, damit die Schrauben bündig mit dem Gehäuse abschließen. Ebenso unproblematisch ist es, die Kabel an die Lautsprecher zu löten, um sie anschließend mit dem Picade PCB zu verbinden. Sinnvoll ist es, den Pluspol jeweils mit farbigem Isolierband zu markieren.

>Schritt-04

Das Gehäuse zusammenbauen

Unsere RaspCade-Schablone enthält bereits Verbindungsstücke, sodass es kein Hexenwerk ist, das Gehäuse zusammenzusetzen. Wir empfehlen, erst einmal auszuprobieren, ob alles passt, bevor Sie zum Kleber greifen, damit Sie genau wissen, welches Teil wo sitzen muss! Verwenden Sie einen hochwertigen Holzkleber und kleben Sie als Erstes eines der Seitenteile auf den Gehäuseboden. Lassen Sie die Teile trocknen.Das kann bis zu 24 Stunden dauern. Danach kleben Sie die zweite Seitenwand an und warten wieder, bis der Kleber getrocknet ist.

>Schritt-05

Abwarten und Tee trinken

Die größte Herausforderung für uns war, darauf zu warten, bis der Kleber so weit abgetrocknet war, dass wir weitermachen konnten. Aber Sie sollten diese Geduld auf jeden Fall aufbringen! Als wir den Gehäusedeckel montierten, haben wir die Seitenteile mit Schnur fixiert und ein (kleines) Gewicht auf dem Deckel platziert, um das Ganze gut zusammenzuhalten, während es trocknet.

>Schritt-06

Die Kabel verbinden

Bevor Sie das Gehäuse weiter montieren, sollten Sie erst einmal die Elektronik-Komponenten an Ort und Stelle bringen. Die Knöpfe und der Joystick werden montiert und die Lautsprecherkabel mit dem Picade PCB verbunden. Dieser wird auf den Abstandshaltern befestigt. Anschließend wird die Vorderfront und das Joystick-Panel moniert. Danach kommt das Display an die Reihe. Jedesmal muss zwischendurch der Kleber trocken! Zu guter Letzt haben wir auf der Rückseite noch einen Magnetverschluss angebracht, um bei Bedarf immer schnell ans Innere des RaspCade zu gelangen.

UMSICHT IST GEFRAGT

Überlegen Sie vorher schon, wo Sie Ihren RaspCade lagern, während der Kleber trocknet, und gönnen Sie ihm Ruhe!

SCHRAUBEN NICHT ZU FEST ZIEHEN

Das Gehäuse ist aus MDF. Ziehen Sie die Schrauben daher vorsichtig an, um das Holz nicht zu beschädigen.

LAUREN EGTS

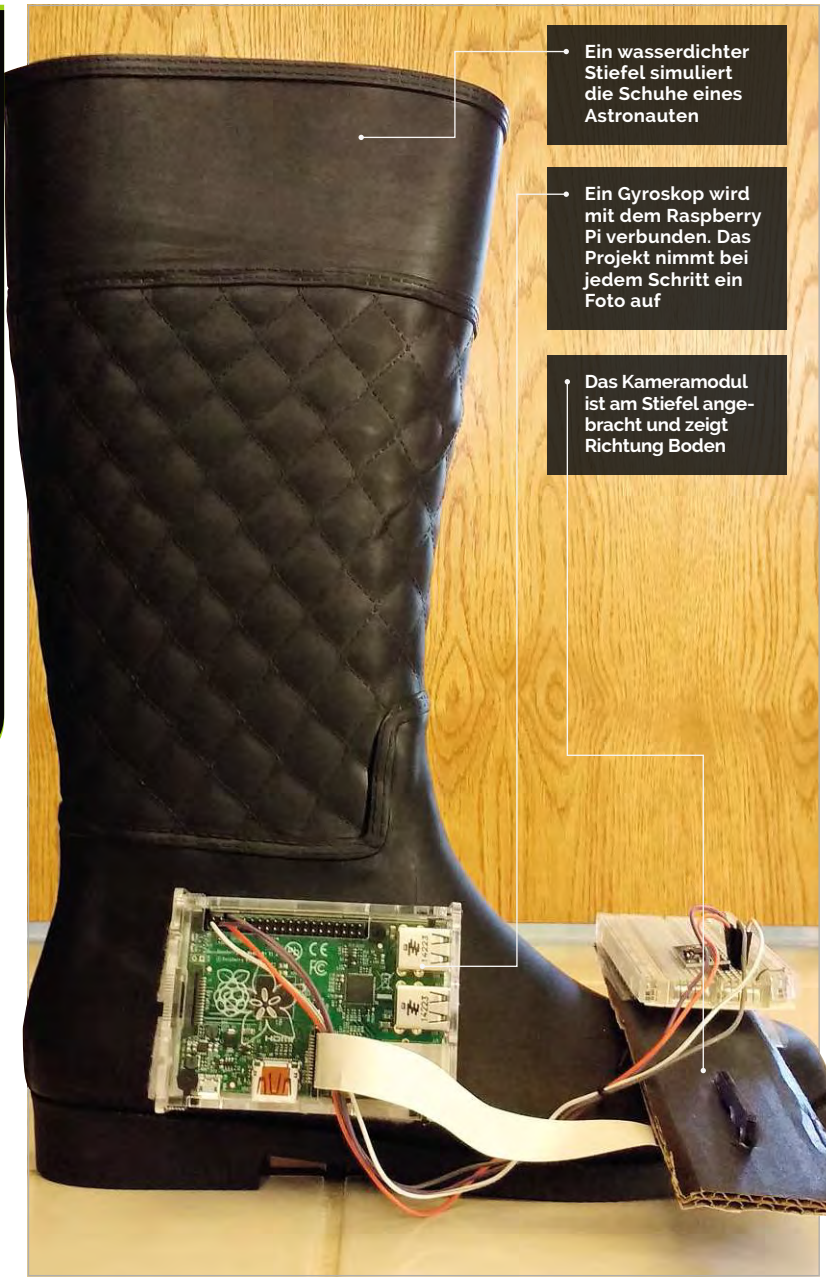
Lauren hat mit 15 ein Praktikum bei der NASA angefangen und arbeitet dort seit drei Jahren.
magpi.cc/2hnYTuJ

PRAKTIKANTIN ENTWICKELT BODENKAMERA

Mithilfe eines Raspberry Pi hat Lauren Egts bei der NASA eine bewegungsgesteuerte Kamera für Bodenaufnahmen entwickelt

Infos

- Neil Armstrong hat die ALSCC im Little Rock Crater genutzt
- Aufgeladen lässt sich BCSI einen vollen Tag nutzen
- Die ALSCC hat stereografische Fotos geschossen
- Der BCSI nimmt digitale Bilder mit einer Auflösung von 1920×1080 auf
- Die ALSCC befand sich etwa 28 cm über dem Boden



Lauren Egts hat die letzten drei Jahre als Praktikantin für die NASA gearbeitet. Sie war an einigen spannenden Projekten beteiligt.

Der BCSI (Boot Cuff Surface Imager) ist ihr neuestes Projekt. Er verwendet ein Pi-Kameramodul und ein Gyroskop, um Nahaufnahmen vom Boden zu machen.

Das Projekt ist eine moderne Variante eines NASA-Geräts, das sich ALSCC (Apollo Lunar Surface Closeup Camera) nennt. Die Kamera hat es zum Mond und zurück geschafft.

„Die ALSCC wurde vom britischen Wissenschaftler Thomas Gold entworfen“, erklärt Lauren. „Er hat sie entwickelt, um die Oberfläche des Mondes betrachten zu können. Die Fotos der ALSCC haben uns aus einer Entfernung von 28 Zentimetern gezeigt, wie die Mondoberfläche beschaffen ist. Das ist, als würde ein Astronaut die Frontscheibe seines Helms auf die Oberfläche legen und sie ansehen.“

Die NASA wollte diese Aufnahmen nicht aus Spaß haben, sagt Lauren. „Damit wurden zusätzliche Informationen über die Mondoberfläche und deren Struktur in Erfahrung gebracht. Mitgebrachtes Material von der Oberfläche konnte diese Informationen nicht liefern, da es zu sehr beschädigt wurde.“

„Die ALSCC besaß rechts und links ein Objektiv – Bilder wur-

Oben Astronauten haben die ALSCC während eines Spaziergangs auf der Mondoberfläche manuell bedient

den gleichzeitig geschossen. Aus diesen 2D-Paaren haben die Wissenschaftler ein 3D-Bild der Mondoberfläche erstelllt."

Auf der Erde konnten Wissenschaftler mithilfe des 3D-Effekts die Mondoberfläche analysieren.

So funktioniert der BCSI

Lauren erläutert: „Das Konzept des BCSI ähnelt dem der Apollo Closeup Camera. Beide nehmen Fotos von der Oberfläche auf." Moderne Bauteile ermöglichen es, den BCSI wesentlich kleiner als eine ALSCC zu bauen. Intelligente Komponenten wie ein Raspberry Pi automatisieren den Prozess des Fotografierens.

„Der BCSI wird auf Höhe des Knöchels am Stiefel befestigt", erklärt Lauren. Ein Gyroskop überwacht die Schritte des Astronauten. Bei jedem Abbremsen schießt der Raspberry Pi ein Foto.

„Natürlich gibt es auf dem Mond oder dem Mars weniger Schwerkraft. Deswegen müssen Parameter entsprechend angepasst werden. Durch Tests habe ich einen Wert ermittelt, der klare Bilder liefert."

Der Weg zur NASA

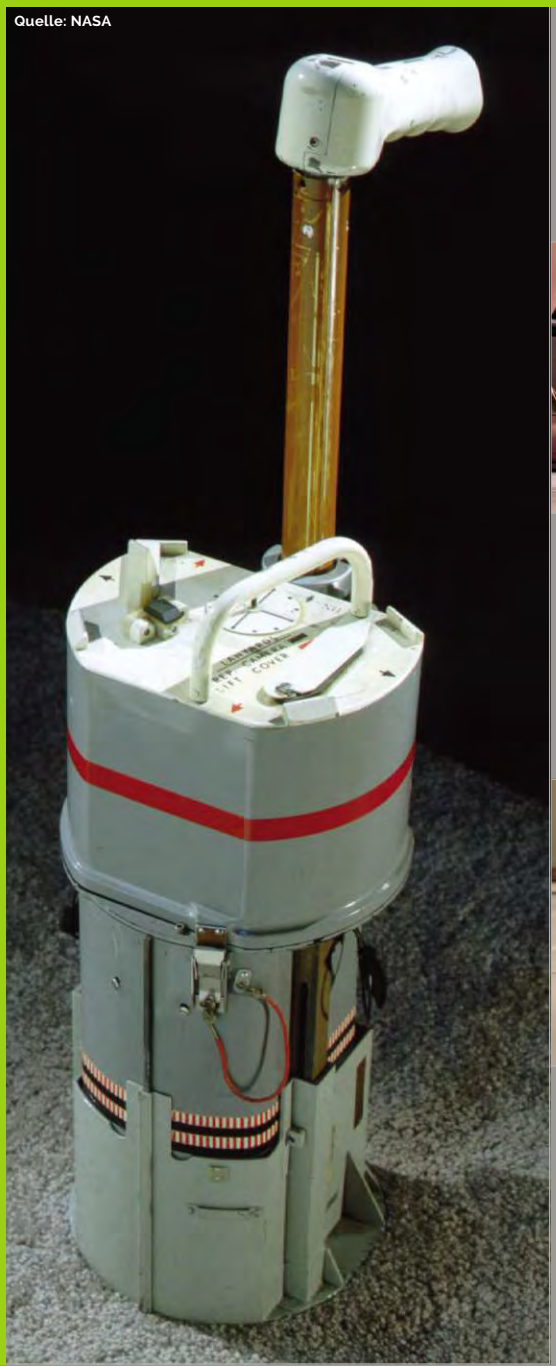
Lauren hat hart gearbeitet, um das Praktikum bei der NASA zu bekommen. „Meinen Mentor, Herb Schilling, habe ich bei einer Mini Maker Faire getroffen. Dort habe ich Scratch präsentiert", erzählt Lauren. „Er hat mich einen Tag zur NASA eingeladen." Nach diesem Tag wurde Lauren gefragt, ob sie im Sommer ein Praktikum für ein paar Wochen machen wolle. Dann hat sie noch zwei weitere Sommer drangehängt.

„Ein Freund meines Mentors hörte von der ALSCC und meinte, eine moderne Variante davon wäre ein interessantes Projekt."

Lauren fährt fort: „Ich wusste ungefähr, wie es aussehen sollte. Deswegen habe ich der Wissenschaft ein Paar Socken geopfert und damit den modernen BCSI entwickelt. Der Sockenprototyp war sehr robust, taugte aber nicht für den Außeneinsatz. Deswegen habe ich mir ein Paar Gummistiefel besorgt und den Boot Cuff Surface Imager dort befestigt."

Lauren macht die Arbeit bei der NASA großen Spaß. „Das Beste am Labor sind die Menschen. Der Altersunterschied bei den Mitarbeitern ist zwar ziemlich groß, aber eine entscheidende Sache verbindet uns alle: Wir lieben Technik."

SO FUNKTIERT DIE BODENKAMERA



Quelle: NASA

>SCHRITT 01

Die ALSCC

Die Originalkamera der NASA kam auf dem Mond zum Einsatz. Es wurden stereografische Fotos geschossen. So ließ sich die Mondoberfläche gut betrachten.

>SCHRITT 02

Der Testsocken

Der BCSI ist eine moderne Variante der ALSCC. Raspberry Pi, Kameramodul und Gyroskop werden am Stiefel des Astronauten befestigt. Das Konzept hat Lauren an einem dicken Socken getestet.

>SCHRITT 03

Jeder Schritt ein Foto

Das Gyproskop kontrolliert, wann der Schritt abbremst. Dann nimmt die Kamera ein Foto des Bodens auf. Das Projekt wurde unter den Bedingungen der Erdanziehungskraft getestet. Lauren hat ermittelt, wie es auf dem Mond oder dem Mars funktionieren würde.

JOHANNES JÄGERS

Der Entwickler von GeRo verfügt über langjährige Erfahrung im Bereich Elektronik und technischem Projekt-Management. **german-robot.com/de**

GERO

Der „German Robot“ GeRo wird von einem Pi angetrieben

Die Kamera spitzt aus der „Nase“ von GeRo heraus

Mithilfe von Aufklebern und hochwertigem D-Druck sieht GeRo richtig gut aus

Sie können ihm auch noch schicke Klamotten anziehen

Infos

- Ein größerer, beweglicherer Roboter ist in Planung
- GeRo war das erste Pi-Projekt von Johannes
- Eine Aufbauanleitung inklusive SD-Image finden Sie online
- Mithilfe von Teilen aus dem 3D-Drucker können Sie das Aussehen von GeRo anpassen
- GeRo hat fünfmal mehr Servos als der Rapiro

Humanoide Roboter faszinieren mich schon immer“, meint Johannes Jägers, der Entwickler von GeRo. „Als ich das Rapiro-Projekt auf Kickstarter entdeckte, beschloss ich, endlich meinen eigenen Roboter zu entwickeln.“

Daraus entstand GeRo, der „German Robot“. GeRo kann laufen und sprechen – und er wird mit einem Raspberry Pi betrieben. Der Roboter lässt sich vom Nutzer programmieren und kann per Webbrowser sogar übers Netz gesteuert werden. Kamera und Mikrofon streamen ihre Daten ebenfalls zu einer mit GeRo verbundenen Webseite.

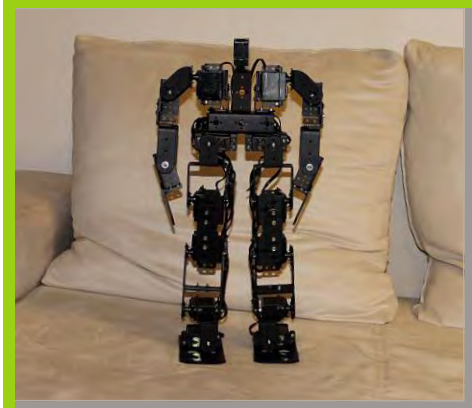
Es ist nicht unbedingt Standard, dass für diesen Typus eines komplexen Roboters ein Raspberry Pi eingesetzt wird. Manche Maker bevorzugen in solchen Fällen eher einen Arduino. „Mir persönlich war ein Computer wie der Pi lieber als ein Arduino”, erklärt Johannes. „Der Pi hat den Vorteil, dass er sehr anschlusssfreudig ist. So lässt sich etwa eine Kamera und ein Mikrofon einbinden. Sie können ein ROS (Robot Operating System) nutzen, Sensordaten erheben und vieles mehr. Außerdem ist der Raspberry Pi sehr preisgünstig – und er bringt eine große Online-Community mit, die Unterstützung bietet, etwa was die Software für die PWM angeht. Diese wird benötigt, um die 17 Servos zu verbinden.“

Den gewünschten Kopf drucken Sie im 3D-Printer

Bei ROS handelt es sich um ein Open-Source-Betriebssystem, das speziell dazu gedacht ist, Roboter zu entwickeln und zu programmieren. Dass GeRo mit ROS läuft, ist eher die Ausnahme bei den RasPi-Robotern: Üblicherweise laufen sie unter Raspbian.

Johannes hat circa zwei Jahre gebraucht, um den Roboter zu ent-

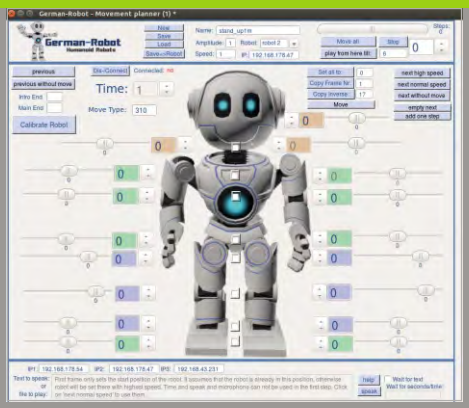
SO STEUERN SIE GERO



>SCHRITT-01

Ab ins WLAN

ist der Roboter fertig aufgebaut, muss er ins WLAN eingebunden werden. So können Sie ihn über ein Interface ansteuern, das auf Ihrem Computer läuft.



>SCHRITT-02

Bedienung über Schieberegler

Sie müsssen nicht für jede Bewegung Code eingeben – GeRo lässt sich über Regler steuern. Über das Interface gelangen Sie auch an die Sprachfunktionen.

>SCHRITT-03

Steuerung vom Mobilgerät aus

Wenn Sie über Ihr Smartphone die entsprechende Seite im Browser anwählen, können Sie den Roboter steuern, ohne exra einen Rechner hochzufahren.

wickeln. Er besteht aus einer Art Metallskelett und verschiedenen Teilen, die im 3D-Drucker gefertigt werden. Darüber hinaus besitzt er eine angepasste Erweiterungsplatine, die unter anderem die Anschlüsse für 17 RC-Servos sowie für einen 6.6-Volt-LiFePo-Akku mitbringt. Eine Aufstellung aller benötigten Teile finden Sie unter **magpi.cc/2fQ7kor**.

„Eigentlich funktioniert GeRo sehr gut“, so Johannes. „Aber damit der Roboter sich bewegt oder spricht, muss er die entsprechenden Befehle erhalten. Ich habe ein Programm geschrieben, das es leichter macht, die einzelnen Servos in die gewünschte Position zu bringen. Trotzdem dauert es etwas, bis alles richtig funktioniert. Das gilt vor allem für die Bewegungen. Der Roboter hat keine Servos, die es ihm erlauben würden, seine Beine in eine andere Richtung zu drehen. Er ist also derzeit noch darauf beschränkt, geradeaus zu laufen. Dafür kann er aber aus jeder Position aufstehen und beliebige Sätze in verschiedenen Sprachen sprechen.“

Johannes hat die Entwicklung von GeRo auf **german-robot.com** zugänglich gemacht. Er schätzt, dass es weniger als 500 Euro kostet, den Roboter zu bauen. Die Konstruktion ist nicht übermäßig kompliziert, aber man sollte dennoch Grundkenntnisse mitbringen, wie Roboter funktionieren. Auch ein gewisse Erfahrung im Löten ist von Vorteil, so Johannes. „Ich fand es wirklich sehr angenehm, mit dem RasPi zu arbeiten, vor allem da er ROS unterstützt. Außerdem kann man über Python und C direkt auf die Hardware zugreifen und auf diese Weise sehr einfach ein komplexes System aufbauen.

Unten Sie brauchen einige Teile, um den Roboter nachzubauen

Prototypen und Aternativen zum Zusammenbau von GeRo

ARBEITEN MIT DEM STECKBRETT

Das unansehnliche Stück Kunststoff mit Löchern ist äußerst nützlich und vielseitig

Sie brauchen

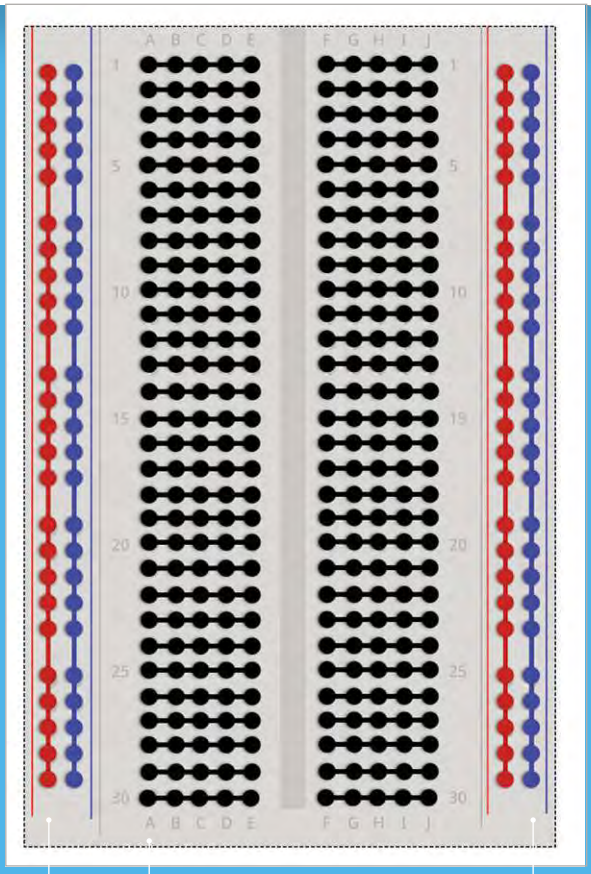
- Steckplatine
- LED
- Widerstand
- Male auf Female Jumperkabel
- Male auf Male Jumperkabel

Um Projekte zu testen, greifen wir häufig auf ein rechteckiges Plastikteil zurück, das als Steckplatine, Steckbrett oder auch Breadboard bezeichnet wird. Nimmt man es ganz genau, lautet die Bezeichnung eigentlich „lötfreie Steckplatine“, da man mit ihrer Hilfe Schaltungen aufbauen kann, ohne zum Lötkolben greifen zu müssen.

Die einzelnen Komponenten werden miteinander verbunden, indem man sie in die Löcher im Breadboard steckt. Die Löcher wiederum sind auch verbunden, und zwar in Form von Reihen (siehe Abbildung rechts). Wenn Sie also ein Kabel oder irgendein anderes Bauteil in ein Loch stecken und ein anderes Bauteil in das Loch daneben, dann hat das den gleichen Effekt, als wären sie direkt miteinander verbunden oder zusammengelötet.

Für viele RasPi-Fans ist es völlig normal, mit einer Steckplatine zu arbeiten. Aber Neueinsteiger müssen sich erst einmal daran gewöhnen: Ein Plastikteil voller Löcher, die in Reihen und Spalten angeordnet sind – das ergibt auf den ersten Blick wenig Sinn.

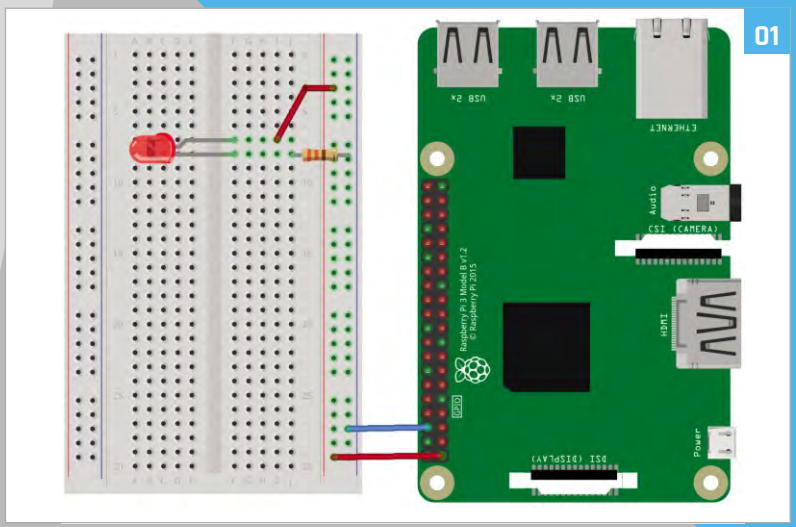
In diesem Workshop zeigen wir Ihnen daher, wie ein Steckbrett funktioniert und wie Sie Ihre erste Schaltung einrichten. Sie werden sehen: Es macht Spaß und bringt Sie unendlich viel weiter, wenn Sie elektronische Bauteile mit Ihrem Pi verbinden können.



Ein Loch in der blauen (manchmal auch schwarzen) Reihe wird mit einem Masse-Pin verbunden.

In die waagrechten Reihen werden die Komponenten gesteckt, aus denen die Schaltung besteht. Hier sind fünf Löcher verbunden.

Die langen senkrechten Reihen werden auch als Verteiler bezeichnet. Sie dienen der Stromversorgung. Die positive Spannung liegt immer auf der roten Reihe.

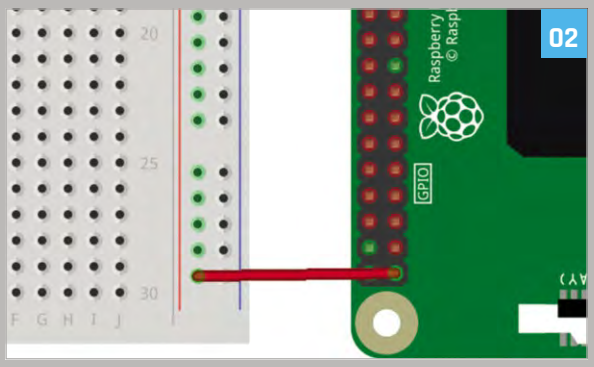


01

>Schritt-01

Fritzing-Diagramm

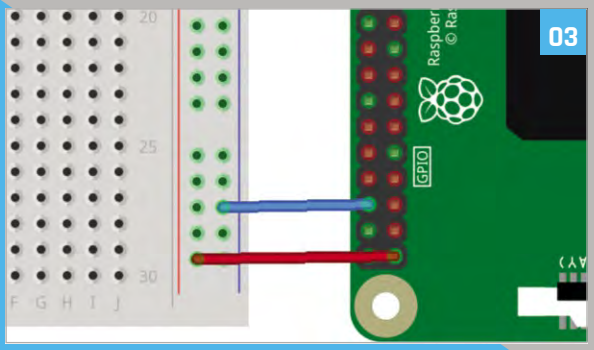
Schaltpläne können für Einsteiger schnell zur Hürde werden. Daher verwenden wir solche übersichtlichen Diagramme. Das Diagramm links nutzt Power- und Masse-Pin des Raspberry Pi, um eine LED zum Leuchten zu bringen.



>Schritt-02

Verteiler für die positive Spannung

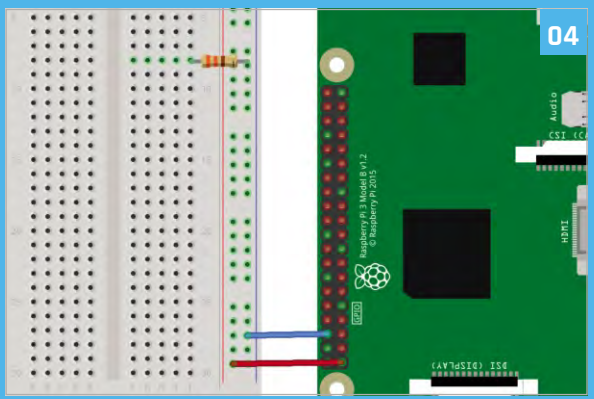
Nehmen Sie ein Female/Male-Jumperkabel (die Farbe des Kabels ist unwichtig) und verbinden Sie das weibliche Ende mit einem 3,3-Volt-Pin des Pi. Die andere Seite stecken Sie in ein Loch auf dem roten Verteiler.



>Schritt-03

Verteiler für die Masse

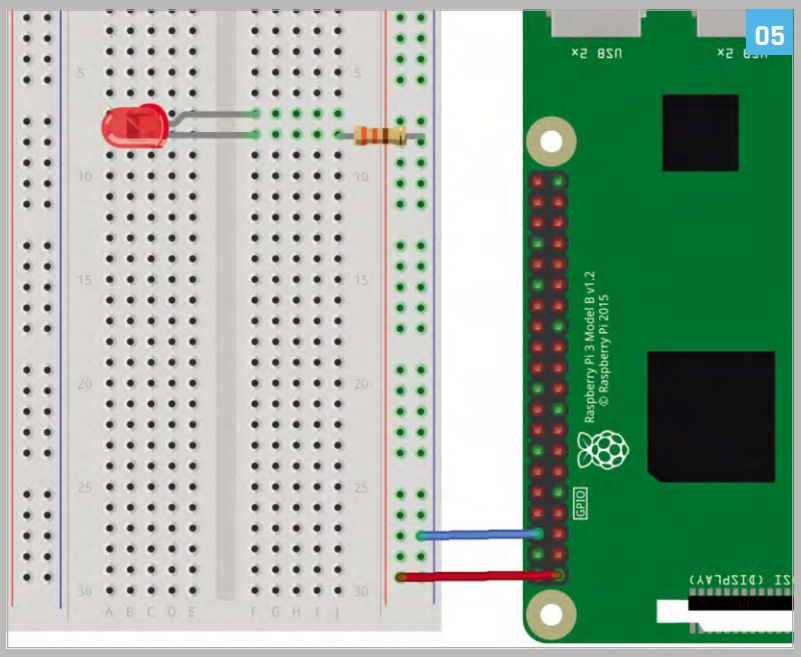
Mit einem weiteren Female/Male-Jumper verbinden Sie einen Masse-Pin (GND) des Raspberry Pi mit einem Loch auf dem blauen Verteiler des Steckbretts. Damit werden alle Löcher der blauen Reihe zu Masse.



>Schritt-04

Einen Widerstand hinzufügen

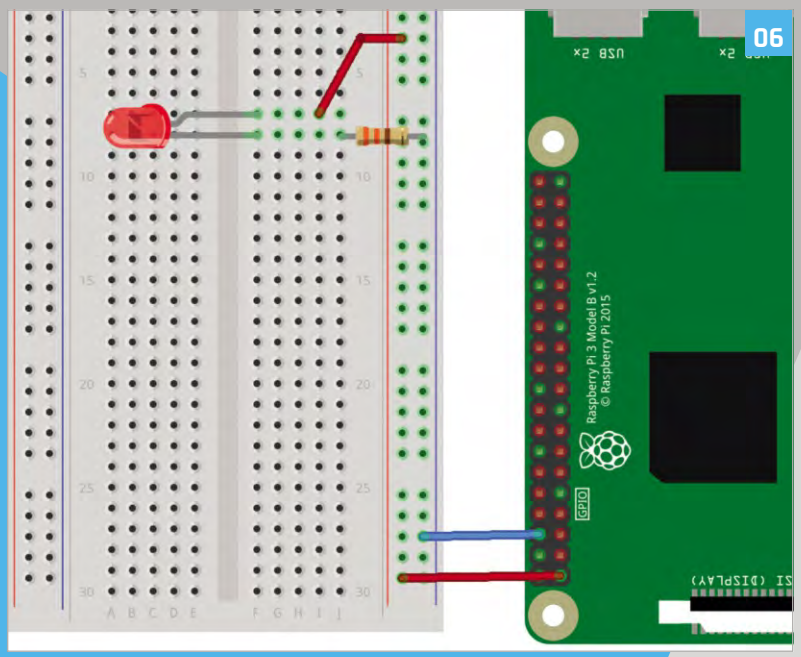
Nehmen Sie einen Vorwiderstand und stecken Sie eines seiner Beine in ein Loch auf dem Masse-Verteiler des Steckbretts. Auf diese Weise ist es nun mit dem Masse-Pin des Pi verbunden. Das andere Bein stecken Sie in ein Loch in einer der waagrechten Reihen des Bretts.



>Schritt-05

Eine LED dazuschalten

Nun nehmen Sie eine LED. Betrachten Sie sie genauer, werden Sie festellen, dass eines ihrer Beine kürzer ist als das andere. Dieses kürzere Bein stecken Sie in ein Loch auf derselben waagrechten Reihe wie den Vorwiderstand. Das Bein ist nun mit dem Widerstand verbunden, dieser wiederum mit dem Masse-Verteiler.



>Schritt-06

Die Verbindung schließen

Stecken Sie das längere Bein der LED in ein Loch in der nächsten waagrechten Reihe. Nehmen Sie einen Male/Male-Jumper und stecken ein Ende in ein Loch neben dem langen LED-Bein, das andere Ende in ein Loch auf dem roten Verteiler. Die LED leuchtet!

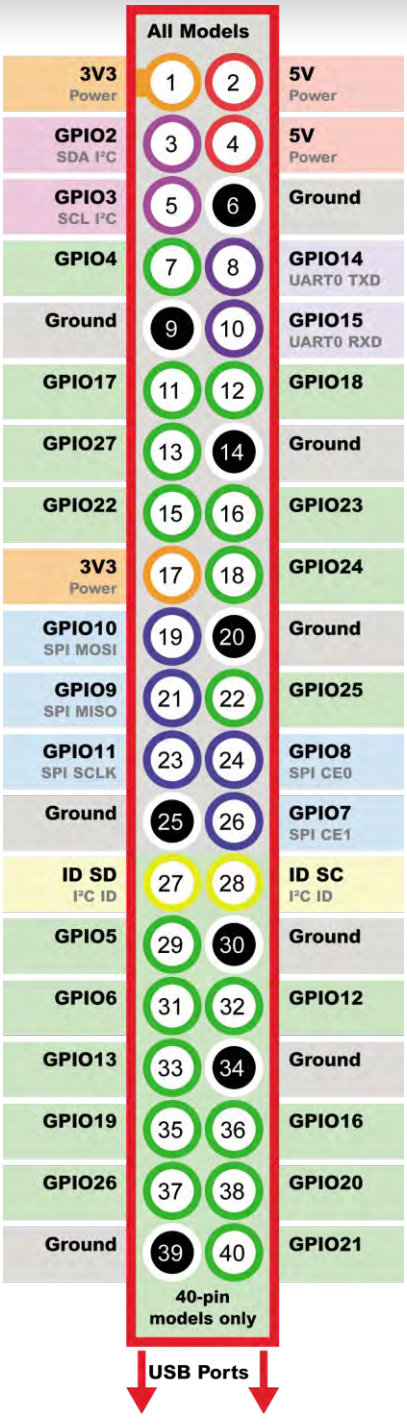
EINSTEIGER-GUIDE:

GPIO ZERO

Verständlich erklärt: So nutzen und programmieren Sie die GPIO-Pins

Pins richtig zuordnen

Bei den Pins auf der 40-poligen Steckerleiste unterscheidet man zwischen der physikalischen Nummerierung und den dazugehörigen Pin-Namen, etwa „Ground“, „5V“ oder „GPIO18“. Beispiel: GPIO18 etwa entspricht der Pin-Nummer „12“, siehe die Grafik rechts. Pin 1 erkennen Sie immer am quadratischen Lötpad auf der Rückseite der Raspberry-Platine. In diesem Tutorial konzentrieren wir uns auf die grün markierten GPIO-Pins.



Sie brauchen

- Raspberry Pi
- Steckplatine
- LEDs
- Taster
- Jumperkabel

Warum ist der Raspberry Pi so beliebt und erfolgreich bei Elektronikbastlern? Grund ist nicht nur der günstige Preis, sondern auch die 40-polige Steckerleiste. Sie erlaubt es nämlich, selbst entwickelte elektronische Schaltungen einzubinden, die Daten von Sensoren auszuwerten oder LEDs, Summer oder Motoren anzusteuern. Diese 40 Pins werden häufig auch als „General Purpose Input Output Pins“ (GPIO) bezeichnet, was aber nicht ganz zutreffend ist, wie die Grafik links zeigt. Nur ein Teil davon sind tatsächlich „echte“ GPIO-Pins.

Apropos: Werfen Sie nochmals einen genaueren Blick auf die Grafik links. Sie sehen: Vier Pins dienen ausschließlich der Stromversorgung; zwei liefern 5 Volt, die beiden anderen 3,3 Volt. Dazu kommen acht Pins, die als Masseanschluss (Ground) dienen.

Besonders interessant sind natürlich die GPIO-Pins, die sich programmieren lassen. Sie können beispielsweise per Python-Befehl an- oder ausgeschaltet werden – praktisch etwa für LEDs. Ein anderes Beispiel: Ändert sich per Tastendruck die Spannung, die an einem Pin anliegt, so könnte dies eine Aktion auslösen. Auf den nächsten Seiten lernen Sie dazu einige Code-Beispiele kennen. Zur Praxis: Im Regelfall baut man seine Schaltung bzw. Prototypen auf Steckplatinen auf, der Flexibilität wegen. Solche Platinen bestehen aus einem Kunststoffbrett mit Federkontakten, die wiederum über Leiterbahnen miteinander verbunden sind (siehe auch Seite 52).

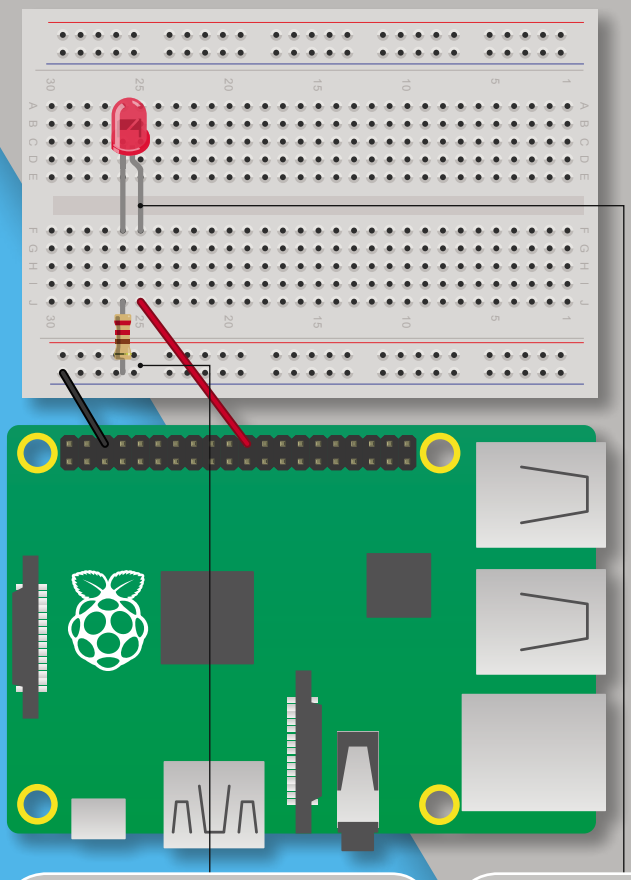
Was die Programmierung angeht: Normalerweise verwendet man Python, Scratch, Java oder andere Sprachen, um GPIO-Pins anzusteuern, Werte abzufragen oder Daten zu übermitteln. Doch das Zusammenspiel zwischen GPIO-Pins und Python ist eine komplexe Angelegenheit, die sich nicht auf Anhieb erschließt. Mit der Bibliothek „GPIO Zero“ ist das anders: Einige simple Befehle reichen, um die GPIO-Pins anzusprechen; zudem ist der Code auch für Einsteiger leichter zu verstehen.

>Schritt-01

Eine Schaltung aufbauen

Das Programmieren der GPIOs ist keine komplizierte Sache. Im Folgenden zeigen wir Ihnen die wichtigsten Schritte – angefangen bei der Verdrahtung der Steckplatine (Grundlagen dazu siehe Seite 52) bis hin zu den ersten Code-Schnipseln. Um Ihnen den Einstieg zu erleichtern, haben wir ein Schaltbild vorbereitet, an dem Sie sich beim Aufbau orientieren sollten (rechts neben diesem Text). Es wurde mit der Software „Fritzing“ erstellt, die Sie von unserer Heft-DVD herunterladen können.

Wir bauen die Schaltung auf, indem wir eine einzelne LED mit dem Anschluss „GPIO25“ (Pin 22) verbinden. Achten Sie darauf, dass das lange Beinchen der LED genauso eingesteckt wird, wie in der Grafik rechts beschrieben. Vertauscht man die Beinchen der LED, fließt kein Strom. Die Schaltung ist so konzipiert, dass die LED aufleuchtet, sobald der GPIO-Pin per Code aktiviert wird. Als Nächstes verkabeln Sie „Ground“ (Pin 6) mit dem Steckbrett, siehe Grafik rechts (schwarzes Kabel). Nehmen Sie den Widerstand und platzieren Sie ein Beinchen in der gleichen Reihe wie das schwarze Kabel. Das rote Kabel verbinden Sie mit GPIO25, siehe Grafik rechts.



Um die LED zu schützen, muss ein Vorwiderstand in Reihe geschaltet werden. Je kleiner der Widerstandswert, desto heller brennt die Leuchtdiode. Beispiel: Starten Sie mit 330 Ω. Die Berechnung erfolgt nach dem Ohmschen Gesetz. Oder nutzen Sie diese Seite: goo.gl/cTN8wi.

Die LED hat zwei Beinchen. Der kürzere Anschluss wird mit dem Widerstand verbunden (also letztendlich mit Masse, Pin 6). Das längere Beinchen ist über das rote Kabel mit GPIO25 auf dem Raspberry-Pi-Board verbunden.

>Schritt-02

Normale Ansteuerung der GPIO-Pins

Wenn wir die LED mit dem regulären Code ansteuern, statt mit der Bibliothek „GPIO Zero“, würden die Code-Zeilen zum Beispiel wie folgt aussehen:

```
import RPi.GPIO as GPIO
GPIO.setmode(GPIO.BCM)
GPIO.setup(25,GPIO.OUT)
GPIO.output(25,GPIO.HIGH)
```

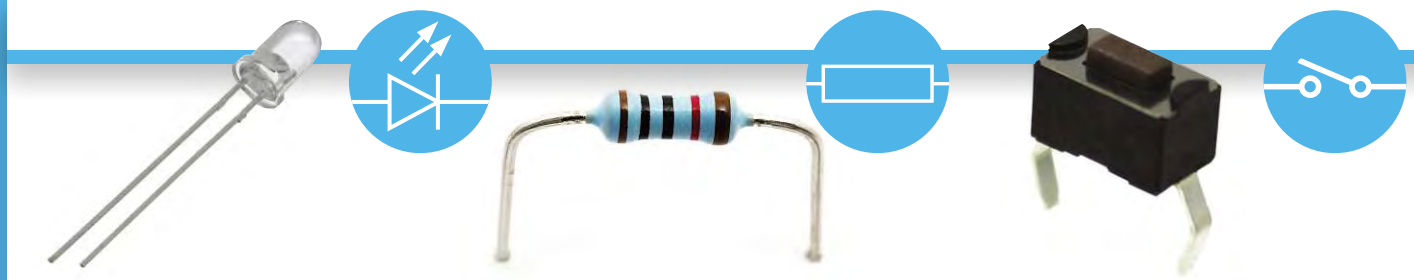
Dieser Code ist zweifellos nachvollziehbar, aber mit Befehlen wie „OUT“ oder „HIGH“ sicherlich nicht intuitiv verständlich. Setzt man alternativ die GPIO-Zero-Bibliothek ein, erscheint vieles klarer, wie das Code-Beispiel im nächsten Schritt zeigt.

>Schritt-03

Pin-Aktivierung mit GPIO-Zero

GPIO Zero ist eine Python-Bibliothek, die stark vereinfachte Schnittstellen bereitstellt – ideal für alle Raspberry-Pi-Einsteiger. Sehen Sie selbst:

```
from gpiozero import LED
led = LED(25)
led.on()
```



LED
Die Beinchen von LEDs sind unterschiedlich lang. Das ist beim Aufbau der Schaltung immer zu beachten.

WIDERSTAND
Um die LEDs zu schützen, müssen in der Schaltung immer passende Vorwiderstände eingeplant werden.

TASTER
Sie vervollständigen die Schaltung. Taster lassen sich gut per Code auswerten und können Aktionen starten.

Schreiben Sie die obigen Code-Zeilen in die Python Shell. Drücken Sie die **Eingabetaste** nach jeder Zeile.

Die erste Zeile des Programms dient dazu, die `gpiozero`-Bibliothek in Python zu importieren. Die zweite Zeile erzeugt ein LED-Objekt mit der Pin-Nummer als Argument (innerhalb der Klammer). Zeile 3 schaltet den GPIO-Pin mit `led.on()` ein. Mit `led.off()` schalten Sie die LED wieder aus.

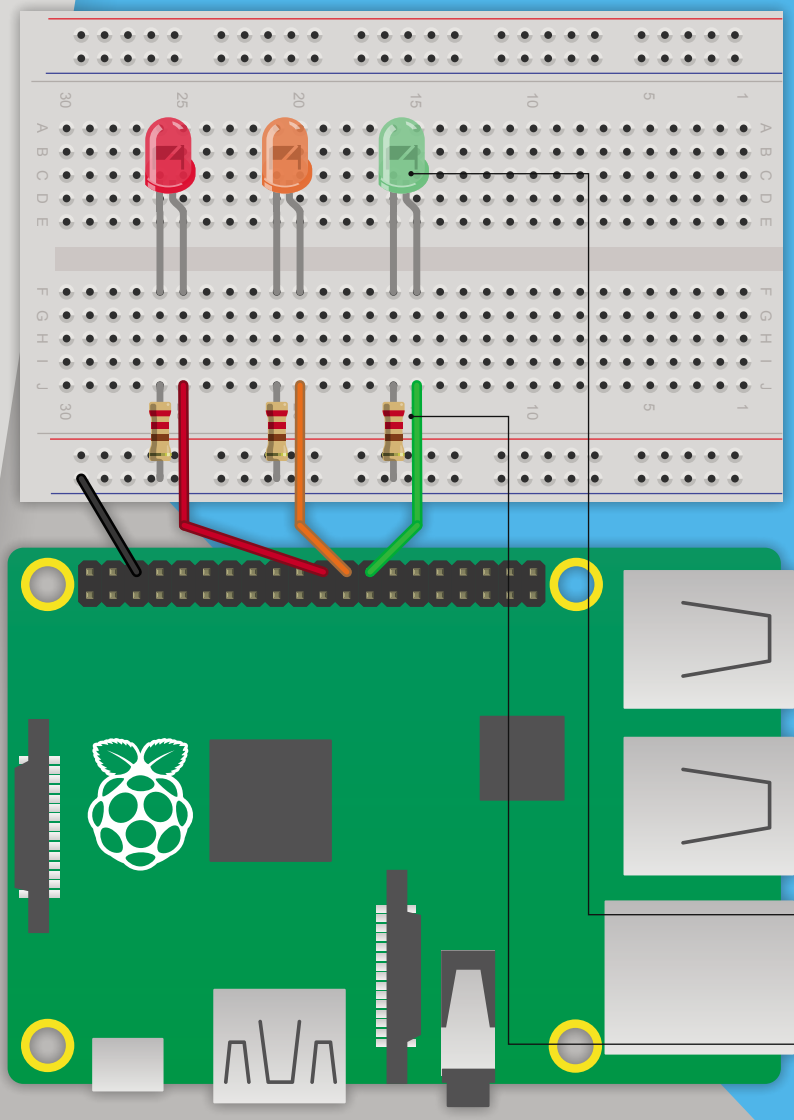
>Schritt-04

Blinkende LED programmieren

Was den Einsatz der GPIO Zero-Bibliothek so interessant macht: Komplexe Aufgaben lassen sich mit simplen Anweisungen lösen. Wenn Sie auf herkömmliche Weise eine blinkende LED programmieren, müssen Sie das `time`-Modul importieren, diverse Befehle für die LED schreiben und mit einem Loop arbeiten. Mit GPIO Zero erledigen Sie alles viel eleganter:

```
led.blink()
```

Schon blinkt das Licht im Sekundentakt. Mit dem Befehl `led.off()` stoppen Sie den Ablauf.Fertig!



Das kurze Ende der LED ist mit einem Widerstand verbunden, das längere Beinchen jeweils mit einem eigenen GPIO-Pin. Diese Pins schalten die LEDs der Farbampel.

Alle Widerstände sind an der Masse-Schiene angeschlossen. Sie ist mit dem GND-Pin (6) auf dem Raspberry Pi verbunden.

>Schritt-05

Werte vorgeben

GPIO-Zero-Befehle sind praktisch: Man kann Argumente per Klammer übergeben. Beispiel: Schreiben Sie `led.blink(` und legen Sie eine Pause ein. Eine gelbe Box mit folgendem Text erscheint: „`on_time=1, off_time=1, n=none, background=True)`“.

Dies sind die Parameter, die Ihnen zur Verfügung stehen, wenn Sie `blink` genauer spezifizieren wollen. Sie geben damit die Zeitdauer in Sekunden vor, wie lange die LED leuchtet, wie häufig sie blinkt oder wann die LED ausgeschaltet wird. Zusätzlich können Sie weiteren Code ausführen, während die LED blinkt.

Die Standardwerte nach jedem Argument lauten: „one second on“, „one second off“, „none“ (die LED blinkt fortlaufend, es sei denn, Sie geben etwas anderes vor) sowie „True“ (an dieser Stelle könnten Sie weiteren Code ausführen lassen). Wenn Sie bestimmte Werte vorgeben möchten, fügen Sie eine Ziffer für jeden Wert hinzu (von links nach rechts):

```
led.blink(4,2)
```

Die LED leuchtet für vier Sekunden und bleibt zwei Sekunden lang dunkel. Mit `led.off()` beenden Sie das Blinken. Ein weiteres, alternatives Beispiel:

```
led.blink(n=3)
```

Die LED blinkt dreimal, dann ist Schluss.

>Schritt-06

Eine Ampel-Simulation

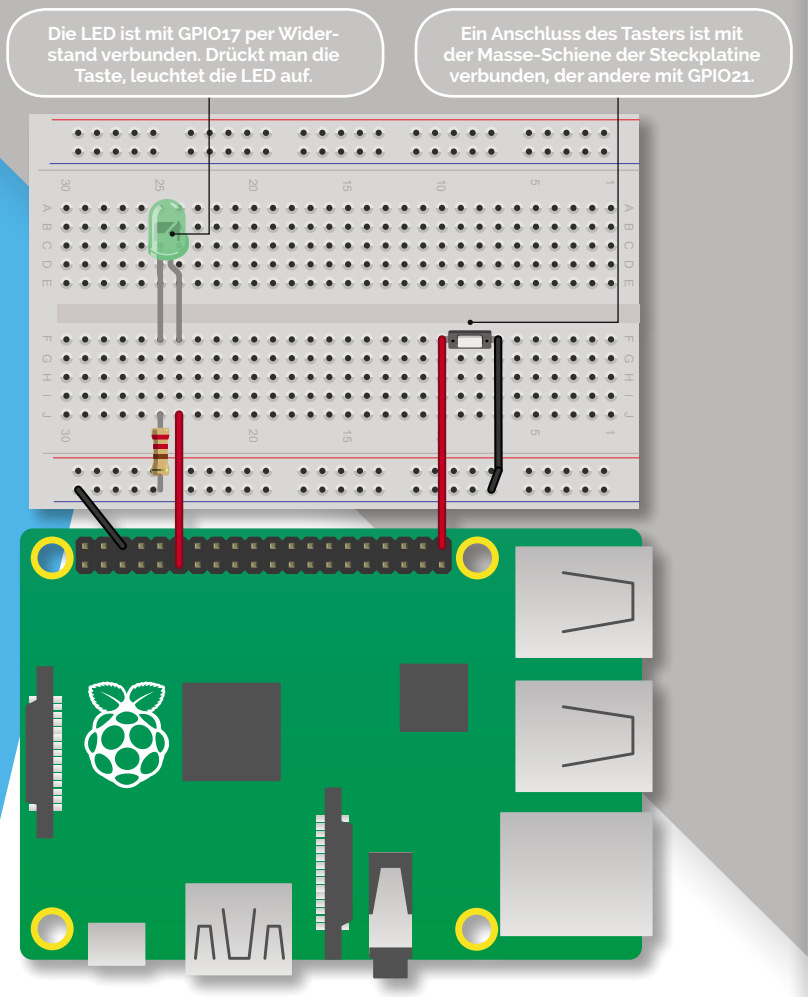
Nachdem Sie die Grundlagen kennen, erweitern wir die Schaltung mit zwei weiteren LEDs zu einer Ampel in den Farben Rot, Gelb und Grün.

Verbinden Sie die beiden LEDs jeweils mit einem Widerstand (gleicher Ohm-Wert, siehe Grafik). Eine Widerstandsfarbcode-Tabelle finden Sie auf dieser deutschsprachigen Seite: **goo.gl/eZZ5dO**. Das lange Beinchen der gelben LED wird mit GPIO8 verbunden, das lange Beinchen der grünen LED mit GPIO7. Beide Pins liegen unmittelbar neben GPIO25.

Jetzt brauchen wir ein Programm, um die Ampel zu steuern. Starten Sie Python 3 und wählen Sie dann **File | New File**. Laden Sie **traffic_lights.py** und starten Sie das Programm mit **Run | Run Module** (alternativ mit **F5**). Sobald das Python-Programm läuft, sehen Sie die Ampel in Aktion.

Sprache
>PYTHON

DOWNLOAD:
magpi.cc/2ehTqVq



>Schritt-07

Taster hinzufügen

Als Nächstes erweitern wir die Schaltung mit einem Taster und verbinden ihn mit GPIO21. Die LED hängt diesmal an GPIO17. Wie bei den übrigen Beispielen benötigen Sie dazu entsprechende Kabel. Bei den beiden Anschlüssen des Tasters gehen Sie so vor: Das eine Beinchen wird mit dem GND-Pin (siehe Grafik oben, schwarzes Kabel) also Ground/Masse verbunden; das andere Beinchen benötigt eine Leitung (siehe rotes Kabel) zum GPIO21-Pin.

>Schritt-08

Tastendruck auswerten

Im Gegensatz zu LEDs benötigen Schalter und Taster im Regelfall keine Vorwiderstände. Sie haben auch keine Polarität, deshalb ist keine Unterscheidung durch unterschiedlich lange Beinchen nötig.
Drückt man den Taster, wird der Stromkreislauf geschlossen, was wiederum eine entsprechende Reaktion des Programms auslöst, sofern es geladen ist.
Starten Sie Python 3 auf Ihrem Raspberry Pi. Legen Sie eine neue Datei an. Übernehmen Sie den Code von **button.py**, sichern Sie die Datei und starten Sie das Programm. Sobald Sie den Taster auf der Steckplatine drücken, leuchtet die LED auf.

button.py

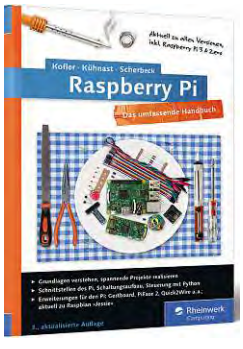
```
from gpiozero import LED, Button
from signal import pause

led = LED(17)
button = Button(21)

button.when_pressed = led.on
button.when_released = led.off

pause()
```

traffic_lights.py

```
from gpiozero import LED
from time import sleep

red = LED(25)
amber = LED(8)
green = LED(7)

green.on()
amber.off()
red.off()

while True:
    sleep(10)
    green.off()
    amber.on()
    sleep(1)
    amber.off()
    red.on()
    sleep(10)
    amber.on()
    sleep(1)
    green.on()
    amber.off()
    red.off()
```

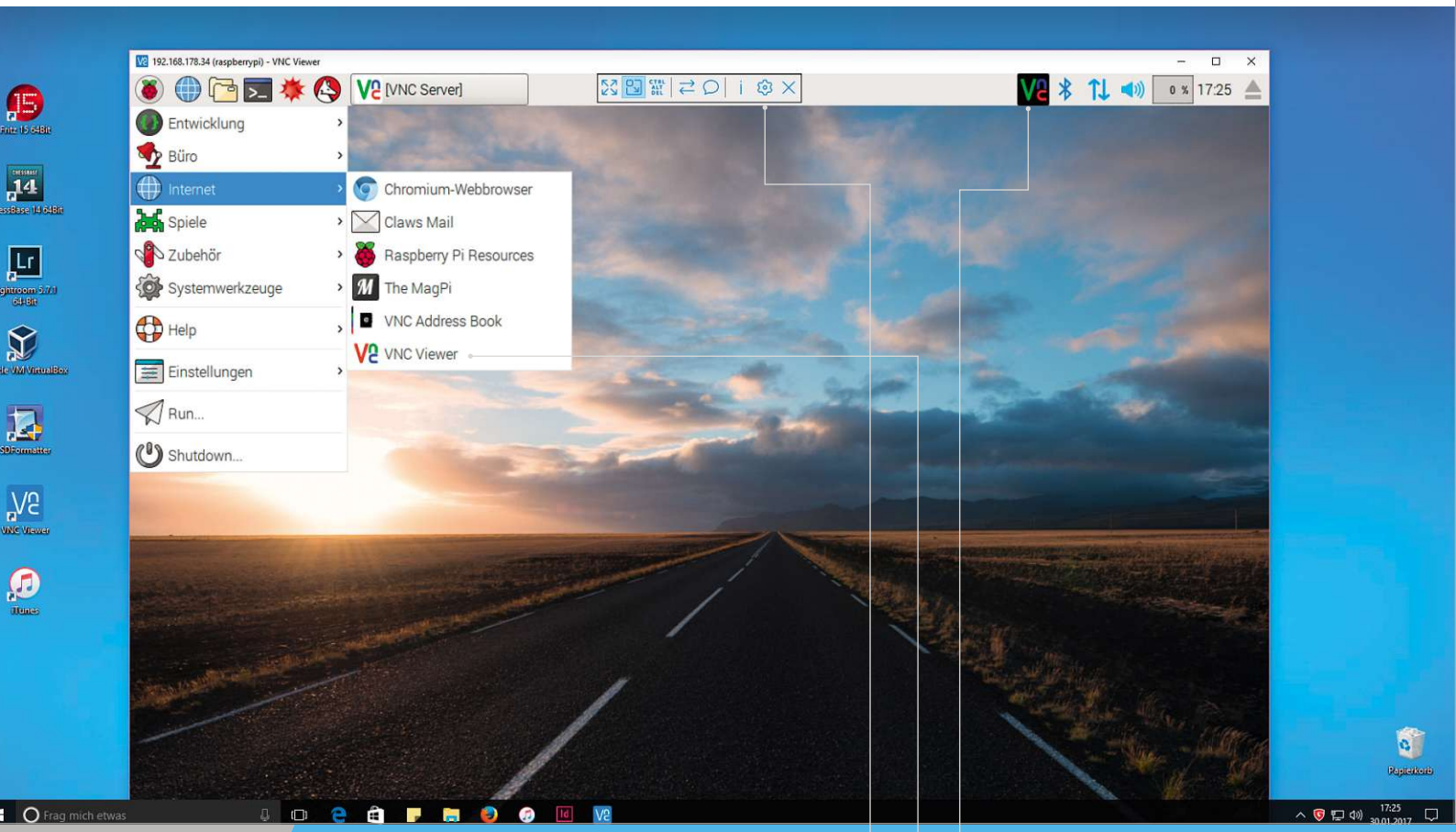
Wissen vertiefen

Wenn Sie eigene Hardware-Projekte von Grund auf konzipieren, eigene Schaltungen entwickeln oder sich intensiv mit GPIO-Pins beschäftigen wollen, empfehlen wir Ihnen das rund 1.000-Seiten starke Werk *„Raspberry Pi: Das umfassende Handbuch“*. Es deckt auch Themenbereiche wie Elektronik oder Sprachen wie Python und PHP sowie die Automation mit Shell-Scripts ab.
ISBN: 978-3836242202, Preis: 39,90 Euro



EINSTEIGER-GUIDE ZU VNC

So steuern Sie Ihren Raspberry Pi von anderen Rechnern aus



Sie brauchen

- Raspberry Pi
- Raspbian
- Laden Sie VNC Viewer für Ihren Computer oder Smartphone

Sie wollen Ihren Raspberry Pi aus der Ferne steuern, etwa von einem Windows-PC, einem Linux-Rechner, dem Mac oder einem Smartphone aus? Dann haben wir im Folgenden die perfekte Lösung für Sie parat: Mit VNC (Virtual Network Computing) greifen Sie direkt auf den Pi zu – als würden Sie unmittelbar davorsitzen.

Ein weiterer Vorteil von VNC: Sie können auf den zweiten Monitor, eine zusätzliche Maus sowie eine Tastatur verzichten, denn die grafische Benutzeroberfläche, in diesem Fall Pixel, wird eins zu eins auf den Windows-Rechner oder ein Tablet übertragen. Falls Sie Ihren RasPi also gerne mit der Maus steuern – kein Problem, VNC macht es möglich.

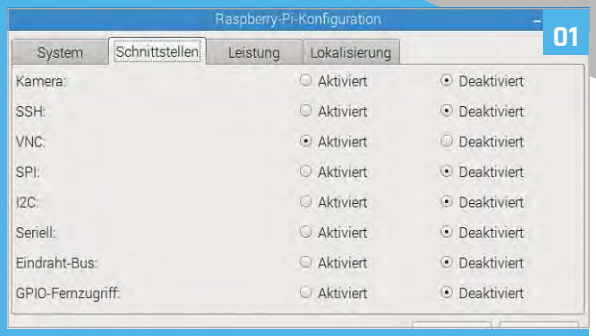
Pixel,die neueste Desktop-Umgebung für Raspbian, hat bereits einen VNC-Server an Bord. Er muss im Prinzip nur noch aktiviert werden. Von diesem Zeitpunkt an startet der VNC-Server automatisch, sobald Sie den Raspberry Pi einschalten.

Noch ein wichtiger Hinweis: Die einzelnen Datenpakete werden bei VNC unverschlüsselt übertragen. Lediglich das Session-Passwort ist beim Transfer geschützt. Für die meisten Bastelprojekte im privaten Bereich dürfte das aber keine Rolle spielen.

Der VNC-Server lässt sich über das Menü im Pixel-Desktop konfigurieren.

In der Toolbar wählen Sie Optionen wie Zugriffsart, Bildskalierung oder Full-Screen-Modus.

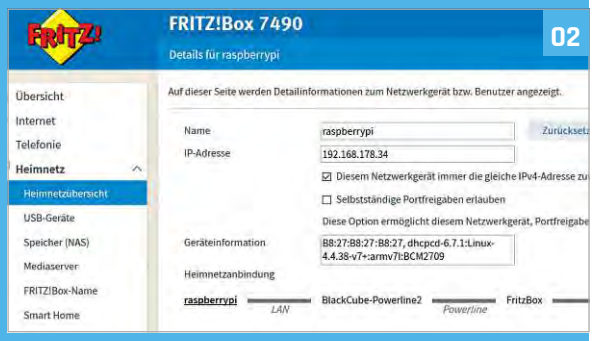
Sie können nun per VNC-Viewer von jedem Betriebssystem aus auf Ihren Raspberry Pi zugreifen.



>Schritt-01

VNC aktivieren

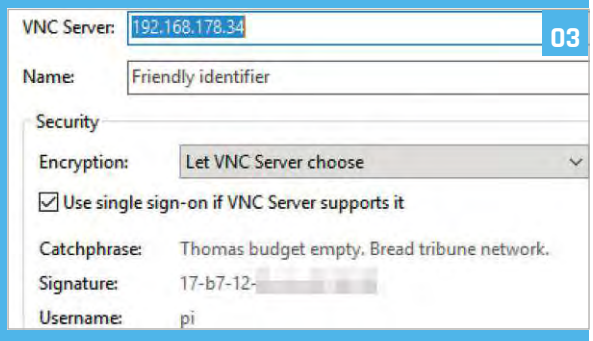
Starten Sie den Pi. Klicken Sie im Menü auf **Einstellungen | Raspberry-Pi-Konfiguration | Schnittstellen**. Setzen Sie „VNC“ auf **Aktiviert** und bestätigen Sie mit **OK**.



>Schritt-02

IP-Adresse festlegen

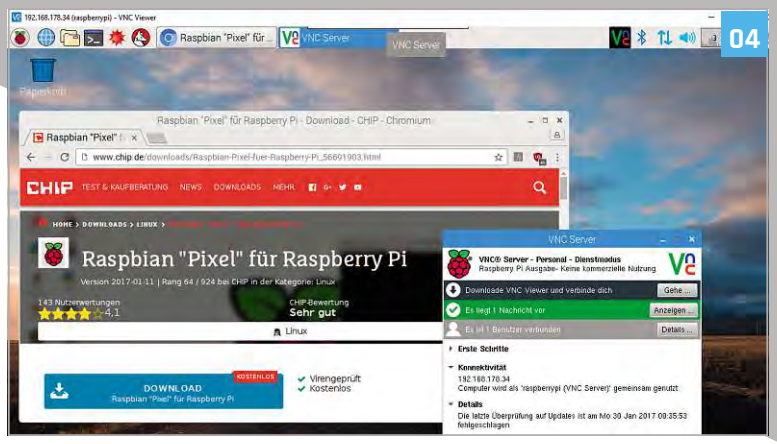
Sie brauchen die IP-Adresse des Pi, um den VNC-Viewer konfigurieren zu können. Bei dem weit verbreiteten Router Fritzbox finden Sie diese Adresse unter **Heimnetz | Heimnetzübersicht | Detail**. Weisen Sie dem Pi diese Adresse fest zu (Häkchen bei dieser Option setzen). Bei anderen Routern funktioniert es ähnlich.



>Schritt-03

VNC-Viewer laden

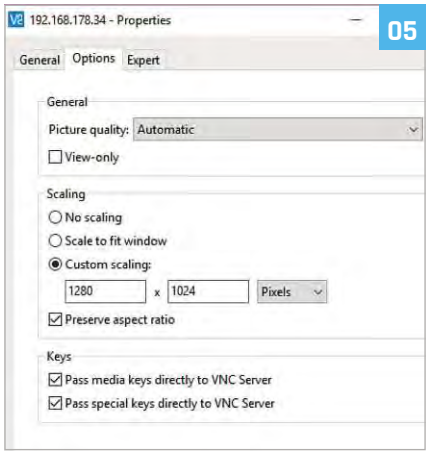
Laden Sie den VNC-Viewer von der **Heft-DVD**. Folgen Sie bei der Installation dem Assistenten. Sobald alles erledigt ist, starten Sie den VNC-Viewer und tragen die IP-Adresse des Raspberrys ein (siehe Schritt 02). Unter **File | Properties** aktivieren Sie bei „Encryption“ die Option „Let VNC Server choose“. Bestätigen Sie die Eingaben mit **OK**.



>Schritt-04

RasPi-Desktop übernehmen

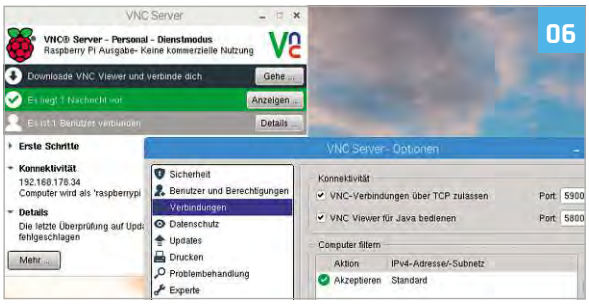
Tragen Sie **pi** als Name ein und wählen Sie dann als Passwort **raspberry** (es sei denn, Sie haben diese Angaben auf Ihrem RasPi geändert). Bestätigen Sie mit **OK**. Auf Ihrem Rechner sollte wenige Sekunden später der Raspbian-Pixel-Desktop erscheinen. Von diesem Moment an lässt sich der Raspberry Pi vom entfernten Computer oder Smartphone aus steuern.



>Schritt-05

Konfiguration ändern

Im virtuellen Fenster (entfernter Rechner) sehen Sie oben einen schmalen Streifen, wenn Sie über die Fensterleiste streichen – es ist die Toolbar. Hier können Sie zum Beispiel den Full-Screen-Modus aktivieren oder den Skalierungsfaktor des Desktops ändern oder Dateien zum RasPi transferieren.



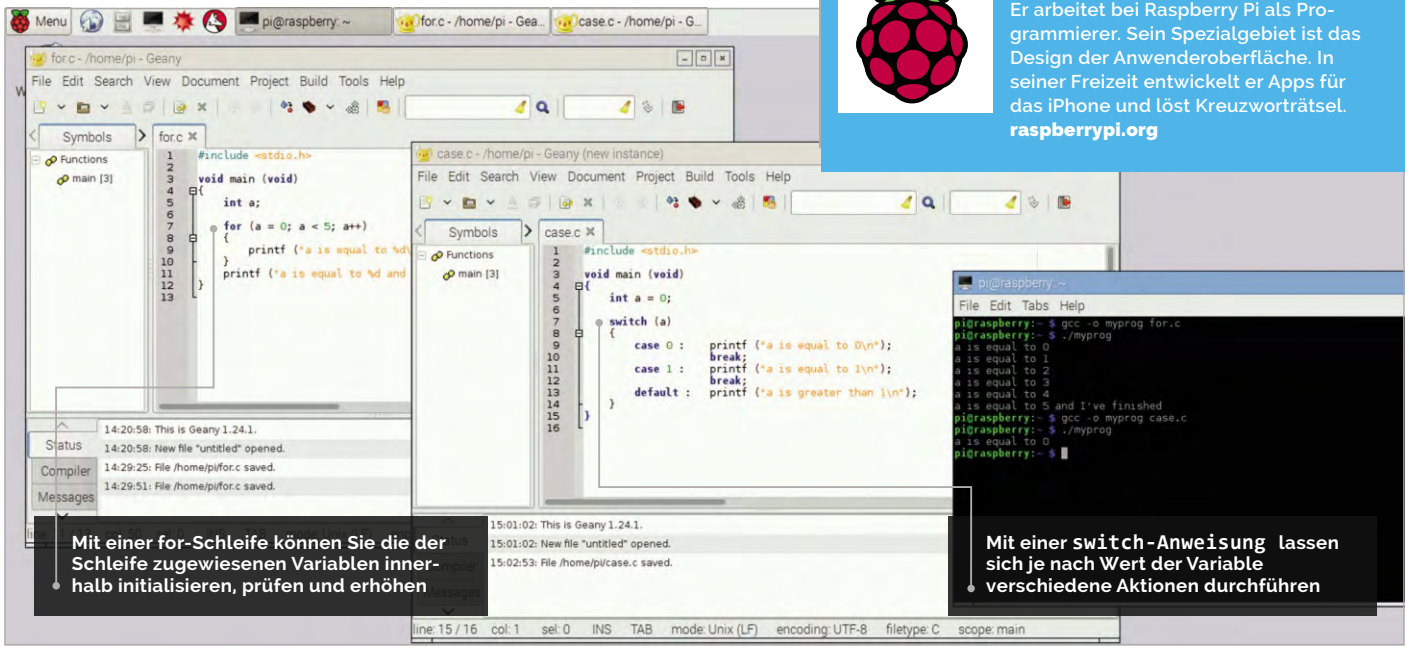
>Schritt-06

VNC-Server konfigurieren

Nicht nur die Konfiguration des VNC-Viewers auf Ihrem Windows- oder Linux-Rechner lässt sich ändern. Auch der VNC-Server, der auf dem Raspberry Pi läuft, bietet Ihnen diverse Konfigurationsvarianten. Dazu klicken Sie oben im Fenster des Pixel-Desktops mit der rechten Maustaste auf das VNC-Icon und wählen **Optionen**. Im folgenden Fenster nehmen Sie bei Bedarf die jeweiligen Änderungen vor.

SIMON LONG

Er arbeitet bei Raspberry Pi als Programmierer. Sein Spezialgebiet ist das Design der Anwenderoberfläche. In seiner Freizeit entwickelt er Apps für das iPhone und löst Kreuzworträtsel.
raspberrypi.org



TEIL 04

EINFÜHRUNG IN C
FLUSSKONTROLLE

Für Schleifen und case-Anweisungen gibt es elegante Kontrollmechanismen

Die **if**-Anweisung und die **while**-Schleife aus dem dritten Teil sind sehr einfache Kontrollmechanismen. In dieser Ausgabe sehen wir uns etwas komplexere Strukturen an. Auf diese Weise wird der Code kürzer und effizienter.

Die **while**-Schleife ist sehr nützlich, aber viele Programmierer bevorzugen die **for**-Variante. Der Grund ist, dass die logische Kontrolle der Schleife komplett an einem Ort zusammengefasst ist. Ein Beispiel:

```
#include <stdio.h>

void main (void)
{
   int a;
   for (a = 0; a < 5; a++)
   {
      printf ("a entspricht %d\n", a);
   }
   printf ("a entspricht %d und Ich
      bin fertig", a);
}
```

Das Konstrukt unterscheidet sich gar nicht so sehr von einer **while**-Schleife. Allerdings ist der gesamte Kontrollmechanismus innerhalb der runden Klammern platziert, nach dem Schlüsselwort **for**. Sie sehen an dieser Stelle drei Anweisungen, die durch Strichpunkte getrennt sind. Entsprechend der Reihenfolge sind das Ausgangszustand, Prüfung und Inkrement.

a = 0 ist der Ausgangszustand. Hier wird **a** am Anfang der Schleife als **0** initialisiert.

a < 5 ist die Prüfung. Das ist äquivalent zur **while**-Schleife. Hier wird nach jedem Durchlauf der Schleife geprüft. Liefert der Test „wahr“ zurück, wird der Code danach ausgeführt. Sobald die Prüfung falsch ist, wird der Code nach den geschweiften Klammern am Ende des Schleifen-Codes ausgeführt.

a++ ist das Inkrement. Der Code wird nach jedem Durchlauf der Schleife und vor der nächsten Überprüfung ausgeführt. In unserem Fall wird die Variable **a** immer um die Zahl 1 erhöht.

Was passiert also in dieser **for**-Schleife? Zunächst wird **a** auf 0 gesetzt. Danach prüft der Code, ob **a** kleiner als 5 ist. Das ist wahr und das Programm führt den Code in den Klammern aus, der den Wert von **a** ausgibt. Anschließend wird **a** um 1 erhöht.

Die Schleife wird so lange wiederholt, bis die Prüfung unwahr ist. In Sachen Ergebnis sind **for**- und **while**-Schleifen ziemlich identisch.

Switch-Anweisungen

Sehr häufig kommt es vor, dass Sie eine Variable mit mehreren Werten vergleichen und entsprechende Aktionen ausführen wollen. Sie können das mit mehreren verschachtelten **if**-Anweisungen realisieren:

LIEBLINGS-SCHLEIFE ...

Alle drei Schleifentypen in C (**while**, **do-while** und **for**) sind für fast jede Situation geeignet, in der eine Schleife gebraucht wird. Verwenden Sie einfach Ihren Favoriten. Einige Leute versteifen sich ausschließlich auf einen Typ. Andere wählen danach aus, welche Schleife jeweils am saubersten aussieht.

```
#include <stdio.h>

void main (void)
{
   int a = 0;
   if (a == 0)
   {
      printf ("a entspricht 0\n");
   }
   else if (a == 1)
   {
      printf ("a entspricht 1\n");
   }
   else
   {
      printf ("a ist groesser als 1\n");
   }
}
```

Das kann allerdings ziemlich umständlich werden. Bei C gibt es eine elegantere Möglichkeit, die sich **switch**-Anweisung nennt.

```
#include <stdio.h>

void main (void)
{
   int a = 0;
   switch (a)
   {
      case 0 : printf ("a entspricht 0\n");
               break;
      case 1 : printf ("a entspricht 1\n");
               break;
      default : printf ("a ist groesser
                  als 1\n");
   }
}
```

Damit erreichen Sie mit weniger Code das gleiche Ergebnis wie mit den **if**-Anweisungen.

Die Anfangszeile enthält das Schlüsselwort **switch**, der Name der Variable ist in runden Klammern. Sie wird dann mit verschiedenen Fällen verglichen. Der Körper in **switch** beinhaltet mehrere **case**-Anweisungen. **a** wird mit jedem Fall verglichen. Entspricht sie dem Wert nach **case**, wird der Code ausgeführt.

Der letzte Fall nennt sich einfach **default**. Jede switch-Anweisung sollte mit **default** abschließen. Der Code wird immer dann ausgeführt, wenn es vorher keine Übereinstimmung gegeben hat.

Beachten Sie, dass die letzte Zeile einer Sektion das Wort **break** ist. Dieses Schlüsselwort teilt dem Compiler mit, dass Sie an dieser Stelle aus der switch-Anweisung „ausbrechen“ wollen. Anders gesagt wird der Code innerhalb von **switch** nicht weiter ausgeführt. Sollten Sie die break-Anweisung vergessen, führt das Programm alle nachfolgenden case-Anweisungen ebenfalls aus. Kompilieren Sie den Code oben und führen ihn aus. Im Terminal sehen Sie, **a entspricht 0**. Entfernen Sie nun die beiden break-Anweisungen:

```
switch (a)
{
       case 0 : printf ("a entspricht 0\n");
       case 1 : printf ("a entspricht
                    1\n");
       default : printf ("a ist groesser
                    als 1\n");
}
```

… und führen Sie das Programm abermals aus:

```
a entspricht 0
a entspricht 1
a ist groesser als 1
```

Das ist so nicht geplant. Dies ist ein häufiger Fehler. Vergessen Sie **break**, führt das zu sehr unerwartetem Verhalten in Ihrem Programm.

Die Schleife vorzeitig verlassen

Die Anweisung **break** dient noch einem weiteren Zweck. Damit beenden Sie **while**- und **for**-Schleifen frühzeitig. Ein Beispiel demonstriert das:

```
#include <stdio.h>

void main (void)
{
 int a = 5;
 while (1)
 {
    printf ("a entspricht %d\n", a);
    a++;
    if (a == 5)
    {
       break;
    }
 }
 printf ("a entspricht %d und Ich
    bin fertig", a);
}
```

Hier haben wir eine **while**-Schleife mit einem Prüfwert von 1. Das ist kein Nullwert und somit immer wahr. Führen Sie Code in den Klammern nach **while(1)** aus, wird er nie enden und Sie haben eine Endlosschleife erzeugt. In unserem Fall beenden wir die Schleife anders. Wir überprüfen den Wert von **a** innerhalb der Schleife mithilfe von **if**. Sobald **a** den Wert 5 annimmt, führen wir **break** aus. Das beendet die Schleife und es geht mit dem Code danach weiter. Ein **break** wie in diesem Fall ist nützlich, wenn Sie die Schleife zum Beispiel im Falle eines Fehlers frühzeitig verlassen wollen.

MEHR INITIALISIERUNGEN

Sie können mehrere Variablen in einer for-Schleife initialisieren. Separieren Sie sie einfach mit Kommas. Bei zwei Variablen sieht das zum Beispiel so aus `for (a = 0, b = 1;<Prüfung>; <Inkrement>)`

MEHRFACHSPRÜNGE

Genau wie bei den Initialisierungen können Sie auch mehrere Variablen in einer Schleife erhöhen. Das wird ebenfalls mit Kommas separiert: `for (a = 0; b = 1; <test> ; a++, b *= 2)`. Praktisch ist das, wenn sich mehrere Variablen dauernd ändern, während die Schleife läuft.

CONTINUE

Das Schlüsselwort `continue` lässt sich als Alternative zu `break` benutzen. Der Befehl springt aber nicht aus der Schleife. Er überspringt lediglich den restlichen Code der momentanen Iteration und der Code landet wieder bei der Prüfung. Damit lassen sich Programme beschleunigen.

HENRY BUDDEN

Als der Raspberry Pi auf den Markt kam, war Henry zwölf. Er hat sich Programmieren und Basteln mit Elektronik selbst beigebracht.
magpi.cc/2eCbaMf

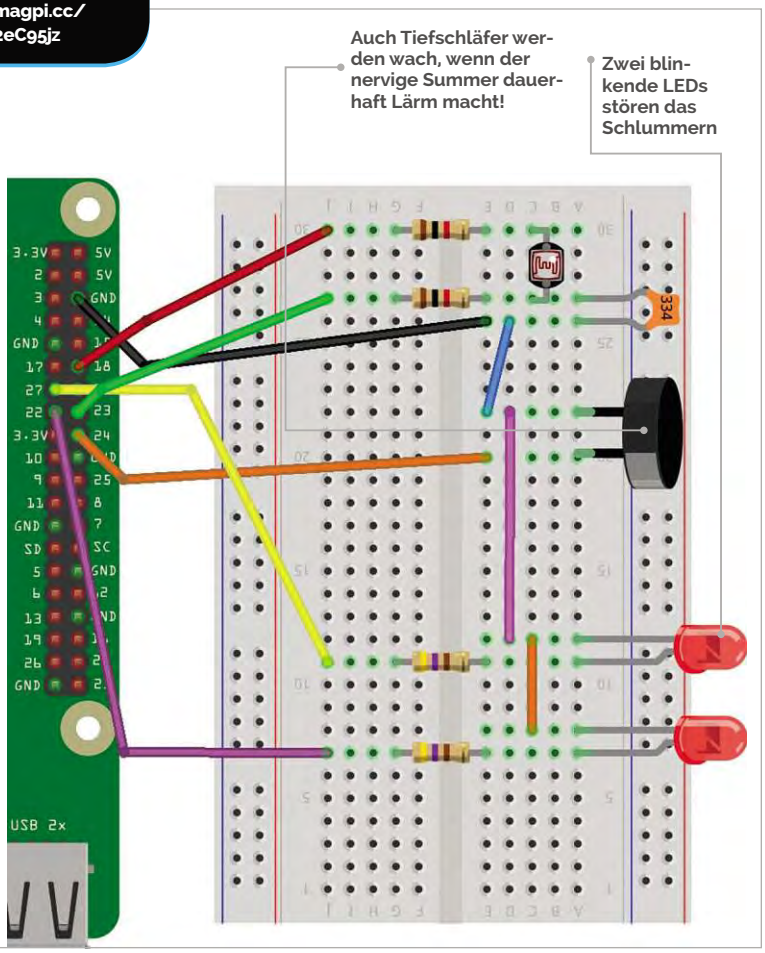
DIY EIGENER WECKER

Mit einer Steckplatine und einfachen Komponenten können Sie das Umgebungslicht messen und einen schrillen Alarm auslösen

Sie brauchen

- 2× LEDs
- Lichtwiderstand (LDR)
- 330 nF Keramikkondensator
- Summer
- 2× 1 kΩ Widerstände
- 2× 470 Ω Widerstände
- 6× Male/Female-Jumperkabel
- 3× Male/Male-Jumperkabel
- Alternativ: Monk Makes Raspberry Pi Electronics Starter Kit **magpi.cc/2eC95jz**

Mit etwas pfiffigem Code und einer Reihe von elektronischen Kleinteilen können Sie auch ohne Wandlerchip das analoge Signal eines Lichtwiderstands mit Ihrem Raspberry Pi auslesen. So lässt sich das Umgebungslicht messen und ein Alarm auslösen. Das Projekt ist eine Erweiterung zu den Projekten, die Sie im „Monk Makes Raspberry Pi Electronics Starter Kit" finden (**monkmakes.com**).



>SCHRITT 01

Widerstände verbinden

Starten Sie mit der Schaltung, indem Sie die Widerstände verbinden. Halten Sie sich dabei an den Schaltplan und stecken Sie die Beine der entsprechenden Komponenten in die Steckplatine. Vergewissern Sie sich, dass die unteren beiden Widerstände 470Ω (gelb, violett und braun) und die beiden oberen 1 kΩ (braun, schwarz und rot) sind.

>SCHRITT 02

Restliche Komponenten anschließen

Nun sind die LEDs an der Reihe. Die langen Beinchen zeigen nach unten, wie Sie im Diagramm sehen. Die flache Seite der LED sollte Richtung 1 kΩ Widerstand gehen. Beim Verbinden des Summers schaut das lange Bein ebenfalls zur unteren Seite der Platine. Zum Schluss verbinden Sie Lichtwiderstand und Kondensator. Dabei spielt die Ausrichtung keine Rolle.

>SCHRITT 03

Mit dem Pi verbinden

Der Raspberry Pi muss ausgeschaltet und vom Stromnetz genommen sein. Benutzen Sie die drei Male-to-Male-Jumperkabel und verbinden Sie die Masse entsprechend dem Schaltplan mit den beiden LEDs, dem Summer und dem Lichtwiderstand. Mit den Male-to-Female-Jumpern verbinden Sie die Steckplatine mit den GPIO-Pins des Raspberry Pi.

>SCHRITT 04

Code herunterladen

Starten Sie den Raspberry Pi. Den Code finden Sie auf unserer **Heft-DVD**. Alternativ öffnen Sie ein Terminal und klonen das GitHub-Repository, das den für den Pi relevanten Code enthält:

```
git clone https://github.com/henrybudden/
rpesk-advanced/
```

Sprache
>PYTHON 2

DOWNLOAD:
magpi.cc/2eCbCtZ

Sunrise_alarm.py

```
from Tkinter import *
import RPi.GPIO as GPIO
import time, math

GPIO.cleanup()

GPIO.setmode(GPIO.BCM)

sunrise = 50

a_pin = 18
b_pin = 23

buzzer_pin = 24
red_pin1 = 27
red_pin2 = 22

GPIO.setup(buzzer_pin, GPIO.OUT)
GPIO.setup(red_pin1, GPIO.OUT)
GPIO.setup(red_pin2, GPIO.OUT)

def discharge():
  GPIO.setup(a_pin, GPIO.IN)
  GPIO.setup(b_pin, GPIO.OUT)
  GPIO.output(b_pin, False)
  time.sleep(0.01)

def charge_time():
  GPIO.setup(b_pin, GPIO.IN)
  GPIO.setup(a_pin, GPIO.OUT)
  GPIO.output(a_pin, True)
  t1 = time.time()
  while not GPIO.input(b_pin):
    pass
  t2 = time.time()
  return (t2 - t1) * 1000000

def analog_read():
  discharge()
  return charge_time()

def read_resistance():
  n = 20
  total = 0;
  for i in range(1, n):
    total = total + analog_read()
  reading = total / float(n)
  resistance = reading * 6.05 - 939
  return resistance

def light_from_r(R):
  return math.log(1000000.0/R) * 10.0

while True:
  GPIO.output(red_pin1, False)
  GPIO.output(red_pin2, False)
  light = light_from_r(read_resistance())
  print light
  x = 0
  if light > sunrise:
    GPIO.output(red_pin1, True)
    GPIO.output(red_pin2, False)
    while True:
      x = x + 1
      GPIO.output(buzzer_pin, True)
      time.sleep(0.001)
      GPIO.output(buzzer_pin, False)
      time.sleep(0.001)
      if x == 250:
        x = 0
        break
    GPIO.output(red_pin1, False)
    GPIO.output(red_pin2, True)
    while True:
      x = x + 1
      GPIO.output(buzzer_pin, True)
      time.sleep(0.001)
      GPIO.output(buzzer_pin, False)
      time.sleep(0.001)
      if x == 250:
        x = 0
        break
```

Sobald die Dateien heruntergeladen und Sie wieder auf der Kommandozeile sind, wechseln Sie in das Verzeichnis mit dem Befehl **cd rpesk-advanced**.

>SCHRITT 05

Code ausführen

Nun testen Sie Ihren Wecker! Vergleichen Sie die Schaltung noch einmal mit dem Diagramm und führen dann den nachfolgenden Befehl aus: **sudo python 01_sunrise_alarm.py**. Sie können den Wecker in einem dunklen Raum testen, indem Sie den Lichtwiderstand mit einer Taschenlampe beleuchten. Die LEDs sollten dann blinken und der Summer Krach machen. Wenn das klappt, haben Sie es geschafft!

>SCHRITT 06

An Ihre Bedürfnisse anpassen

Per Standard wird ein Alarm ausgelöst, wenn der Lichtwiderstand einen Wert von 50 erreicht. Für unseren Test passt das. Damit der Wecker bei Sonnenaufgang losgeht, müssen Sie vielleicht die Datei mit **nano 01_sunrise_alarm.py** anpassen und den Wert in Zeile 13 ändern. Unsere Empfehlung für eine Erkennung der Dämmerung ist 30. Speichern Sie die Änderungen mit der Tastaturkombination **[CTRL]+[X]** und bestätigen Sie die Änderungen je nach Spracheinstellungen mit **[Y]** oder **[J]** gefolgt von der **Eingabetaste**. Führen Sie nun den Code wie in Schritt 5 aus.

JAMES HOBRO

James ist ein Pi-Hacker, der sich besonders für Roboter und Multi-Core-Programmierung interessiert. Er ist außerdem Forscher bei Schlumberger Gould Research in Cambridge.
github.com/jhobro

EINE EINFÜHRUNG IN

PARALLELES PROGRAMMIEREN

So nutzen Sie alle vier Kerne des Pi und erledigen Aufgaben parallel

Schon der Raspberry Pi 2 brachte nicht mehr nur einen Prozessorkern mit, sondern gleich mehrere. Sowohl der Pi 2 als auch der Pi 3 besitzen Quad-Core-Prozessoren. Das ändert die Art, wie wir Programme schreiben, grundlegend, da sich nun Aufgaben parallel erledigen lassen. Im Prinzip erhalten wir vier Mal so viel Geschwindigkeit wie beim ursprünglichen Raspberry Pi. Es ist aber nicht immer ganz einfach, die zusätzliche Rechenleistung optimal zu nutzen. In diesem Artikel stellen wir eine der einfachsten Möglichkeiten für parallele Programmierung vor. Auf diese Weise können Sie die gesamte Rechenleistung des Raspberry Pi einsetzen.

Frühstück mit drei Kernen

Bevor wir uns der Programmierung widmen, sehen wir uns eine banale Alltagsaufgabe an, nämlich die Zubereitung des Frühstücks. Wollen wir den Vorgang als eine Liste von Anweisungen darstellen, könnte der Ablauf vielleicht wie in **Abb 1** aussehen.

Die Frühstücksliste ist ein sequenzielles Programm. Darin befinden sich alle Anweisungen für die Zubereitung des Frühstücks. Irgend etwas stimmt dabei aber nicht. In der Realität würden wir diese Liste niemals als Sequenz, sondern parallel abarbeiten. Natürlich können wir Wasser kochen und gleichzeitig ein Stück Brot toasten. Es lassen sich aber nicht alle Aufgaben simultan erledigen. Sie können so lange keinen Tee zubereiten, bis das Wasser heiß ist.

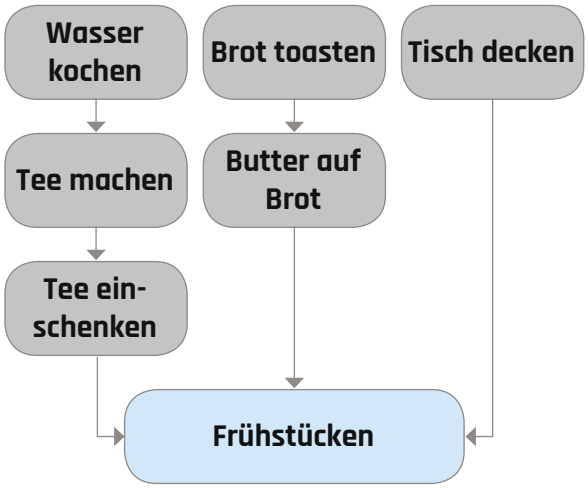
Wie können wir das darstellen? Eine Möglichkeit ist ein Graph, der die Abhängigkeiten der Aufgaben darstellt. Eine entsprechende Aufgabe kann erst dann starten, wenn diejenige abgeschlossen ist, von der sie abhängt. Die Frühstücksliste mit Abhängigkeiten (siehe **Abb 2**) ist eine wesentlich nützlichere Darstellung als das sequenzielle Programm. Sie kommt der Zubereitung des Frühstücks wesentlich näher. Jeder Strang startet mit einer Aufgabe und Sie hangeln sich dann nach unten durch. Sie kümmern sich jeweils immer um die Nachfolgeaufgabe. Sobald das Wasser kocht, können Sie Tee machen.

Natürlich lassen sich nicht alle Programme, die wir parallel laufen lassen möchten, so aufschlüsseln. Ein Abhängigkeitsgraph dieser Art bietet aber eine einfache und leistungsfähige Möglichkeit, viele parallele Aufgabenstellungen zu lösen.

Abb 1 **FRÜHSTÜCK ALS LISTE**

- Wasser kochen
- Tee machen
- Tee einfüllen
- Brot toasten
- Butter auf Brot
- Tisch decken
- Frühstücken

Abb 2 **FRÜHSTÜCKSLISTE MIT ABHÄNGIGKEITEN**



Arbeiten mit Make

Raspbian liefert das Programm Make mit, das einem Abhängigkeitsgraphen folgt. Es führt Aufgaben in der entsprechenden Reihenfolge aus und verteilt sie auf die vier Kerne des Pis. Make wurde für Sprachen wie C oder C++ entwickelt, bei denen Code kompiliert werden muss. Es lässt sich aber auch für die Lösung anderer Probleme nutzen, die man als Aufgabenliste mit Abhängigkeiten ausdrücken kann.

Um Make verwenden zu können, müssen wir ein „makefile“ erstellen, in dem die Abhängigkeiten hinterlegt sind. Als Beispiel generieren wie eine Collage mit Vorschaubildern, die wir aus Originalfotos erzeugen. So etwas benötigen Sie häufig, wenn viele Fotos im Spiel sind. Neben Make benutzen wir außerdem ImageMagick für die Umwandlung der Fotos. Stellen

Sie aus diesem Grund als Erstes sicher, dass die entsprechenden Pakete auf Ihrem Raspberry Pi installiert sind:

```
sudo apt update
sudo apt install make imagemagick
```

Zunächst verwenden wir Make, um die Vorschaubilder aus dem relevanten Original zu erzeugen. Das Ergebnis bildet dann die Collage. Die meiste Arbeit liegt darin, die Vorschaubilder zu kreieren. Der Abhängigkeitsgraph (**Abb 3**) visualisiert den Zusammenhang zwischen Original, Vorschaubild und Collage. Da sich die Vorschaubilder unabhängig voneinander generieren lassen, können wir alle vier Kerne einsetzen. Sobald alle Miniaturansichten fertig sind, lassen wir daraus die Collage erstellen.

Makefile-Regeln

Wir brauchen lediglich das makefile, um die Aufgaben und die Abhängigkeiten zu deklarieren. Zunächst einmal wird die Liste mit den Namen der Originaldateien und den Vorschaubildern definiert. Danach definieren wir die beiden Regeln für die Abhängigkeiten, wie im Graph angegeben. Die erste Regel beschreibt, wie die Miniaturansicht generiert werden soll. In unserem Fall wird jede Datei aus dem Ordner **fullsize** gewandelt. Der Name bleibt gleich, landet aber in einem Verzeichnis **thumbs**. Wir verwenden für das Umwandeln den Befehl **convert**. Beachten Sie, dass in einem makefile Zeilen mit Befehlen mit einem Tab beginnen und nicht mit Leerzeichen. Im Code wird das durch einen langen Unterstrich dargestellt. Die zweite Regel spezifiziert die Collage. Wir verwenden den Befehl **montage** zum Erstellen und zeigen die Collage im Anschluss an. Wir haben Make mitgeteilt, welche Befehle für das Skalieren und das Kombinieren verwendet werden. Eine Reihenfolge haben wir aber nicht angegeben. Make ermittelt das selbständig. Statt eines sequenziellen Programmes haben wir nun eine Aufgabenliste mit Abhängigkeiten. Jetzt arbeitet das System so effizient wie möglich.

Das Programm führen Sie mit diesem Befehl aus:

```
make
```

Make liest nun die Regeln und führt die entsprechenden Aufgaben aus. Per Standard geht Make davon aus, dass nur ein Kern verfügbar ist. Somit wird die Aufgabe sequenziell durchgeführt. Wir können aber angeben, dass mehrere Kerne verfügbar sind:

```
make -j4
```

Mit diesem Befehl spezifizieren wir, dass Make bis zu vier Aufgaben simultan ausführen darf. Das Erstellen der Vorschaubilder ist nun bis zu vier Mal so schnell. Kopieren Sie nun neue Bilder in das Verzeichnis **fullsize** oder aktualisieren Sie existierende Fotos. Lassen Sie Make im Anschluss abermals laufen. Make wird nur die Updates verarbeiten. Das Programm versteht den Abhängigkeitsgraph und kann die Zeitstempel auswerten. Auf diese Weise führt die Software nur notwendige Arbeiten durch. Deswegen können Sie laufende Prozesse auch unterbrechen; die Software wird bei der nächsten Ausführung an der richtigen Stelle weitermachen. Mithilfe des Befehls:

```
make clean
```

löschen Sie alle generierten Vorschaubilder aus dem entsprechenden Ordner sowie die Collage.

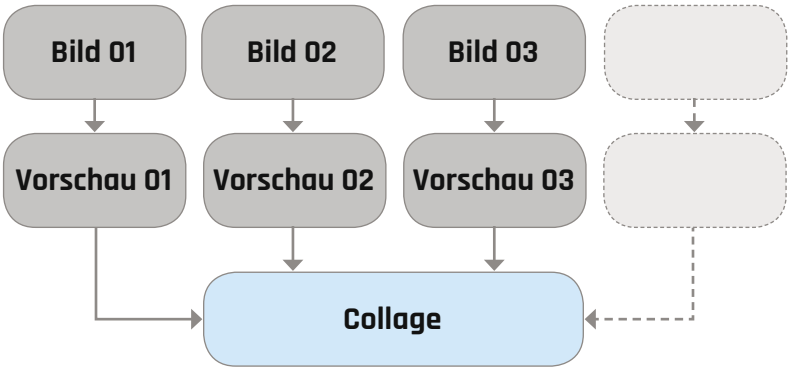
Makefile

Sprache: >BASH

DOWNLOAD: magpi.cc/2bSdSJG

```
# Größe der Vorschaubilder in Pixel
SIZE = 128x128

# Die Liste der Originalfotos (fullsize/* bezieht sich auf alle
# Dateien im Verzeichnis fullsize)
ORIGINALS = $(wildcard fullsize/*)

# Mithilfe der Originalliste die Liste der Vorschaubilder erzeugen
# (Das Programm ändert das Präfix des jeweiligen Originals wird von
# 'fullsize' nach 'thumbs')
THUMBS = $(ORIGINALS:fullsize/%=thumbs/%)

# REGEL 1: Erstelle mit convert von jedem Original ein Vorschaubild.
# Es ist in ImageMagick enthalten. Drehe das Bild, falls nötig.
thumbs/% : fullsize/%
____convert $< -thumbnail $(SIZE) -auto-orient $@

# REGEL 2: Kombiniere alle Vorschaubilder und gib die Collage aus.
montage.jpeg: $(THUMBS)
____montage $(THUMBS) montage.jpeg
____display montage.jpeg &

# Lösche alle Vorschaubilder und die Datei montage.jpeg
clean:
____$(RM) thumbs/* montage.jpeg
```

Abb 3 ABHÄNGIGKEITSGRAPH DER COLLAGE



PIXEL JETZT AUCH AUF DEM COMPUTER

Den **Pixel**-Desktop des Pi gibt es nun auch für ganz normale PCs. Sie können dies mit unserer Heft-DVD gleich einmal gefahrlos ausprobieren

Wäre es nicht klasse, wenn Sie das gleiche Betriebssystem wie auf dem Raspberry Pi auch auf Ihrem Computer laufen lassen könnten? So hätten Sie die gleiche Umgebung auf beiden Geräten, könnten mit denselben Programmen arbeiten und problemlos hin- und herwechseln. Eine spezielle Linux-Distribution der Raspberry Pi Foundation macht dies jetzt endlich möglich: Debian mit Pixel-Desktop.

Natürlich müssen Sie dafür Ihr Standard-Betriebssystem – etwa Windows, Linux oder Mac OS – nicht aufgeben. Sie starten Ihren Rechner einfach komfortabel mit beiliegender Heft-DVD oder legen sich einen bootfähigen USB-Stick mit dem System an. Alle Tools sowie das Betriebssystem finden Sie auf der DVD. Wie Sie starten und das neue Betriebssystem nutzen, zeigen wir im Folgenden Schritt für Schritt. Zudem erfahren Sie, wie Sie einen USB-Stick für den PC vorbereiten.

Noch ein kleiner Hinweis: Debian mit Pixel-Desktop befindet sich noch in der Entwicklung. Daher eignet sich das System noch nicht als dauerhafte Arbeitsumgebung. Zudem gibt es leider noch keine deutschsprachige Distribution. Sie müssen also selbst nachträglich noch das System auf Deutsch umstellen.

START VOM USB-STICK

Wir zeigen Ihnen, wie Sie einen USB-Stick mit Pixel einrichten. Ihr PC startet dann viel schneller als von DVD. Zudem können Sie darauf dann auch speichern (Seite 72).

DER PIXEL-DEKTOP

Der Pixel-Desktop sieht auf Ihrem PC genauso aus wie auf dem Raspberry Pi. Grundlage ist Debian, dasselbe Linux-System, das auch auf dem Pi läuft.

DVD-INHALT

Die Highlights auf der der Heft-DVD zur MagPi 2/2017 sind:
- Debian mit Pixel für den PC (als Live-System und als ISO)
- Raspbian und Noobs
- Toolpack
- Code und Materialien zu den Workshops im Heft

Live-System: Legen Sie die DVD ein und starten Sie Ihren Rechner von DVD. Ggf. müssen Sie dazu das BIOS/UEFI anpassen (siehe Seite 68).

DVD-Start: Führen Sie die Datei »starter.html« im Stammverzeichnis der DVD aus. Sie läuft auf allen Betriebssystemen mit Webbrowser.

DVD kaputt? Sollte diese DVD defekt sein oder fehlen, schreiben Sie bitte eine E-Mail an: dvd@chip.de.

START VON DVD

Einfacher geht es nicht: Starten Sie Ihren PC mithilfe unserer Heft-DVD. Ihr installiertes Betriebssystem wird davon nicht beeinflusst. Falls Sie ein paar Einstellungen und Dateien dauerhaft speichern wollen, zeigen wir, wie das geht.

ERSTER PC-START MIT DER HEFT-DVD

Jetzt geht es los: Legen Sie die Heft-DVD in Ihr Laufwerk und starten Sie den **Pixel**-Desktop auf jedem beliebigen PC

Haben Sie Geduld – das Starten per DVD dauert etwas länger als beim Raspberry Pi.

Debian mit Pixel ist dafür ausgelegt, dass es sich so einfach wie möglich bedienen lässt. Daher verwundert es nicht, dass das System völlig unkompliziert auf Ihrem Rechner läuft – auch wenn dieser normalerweise mit Windows, Linux oder Mac OS betrieben wird.

Booten Sie Ihren Computer einfach mittels der eingelegten **Heft-DVD**. Sofern es im UEFI/BIOS richtig eingestellt ist, lädt dann automatisch Debian mit Pixel statt Ihres regulären Systems. Anschließend können

SIE BRAUCHEN

- Unsere Heft-DVD
- Mac oder PC mit DVD-Laufwerk

Sie damit arbeiten wie auf einem Raspberry Pi. Deutschsprachige Anwender müssen allerdings noch ein paar kleine Änderungen zur Lokalisierung vornehmen – doch dazu später mehr.

DVD einlegen und starten

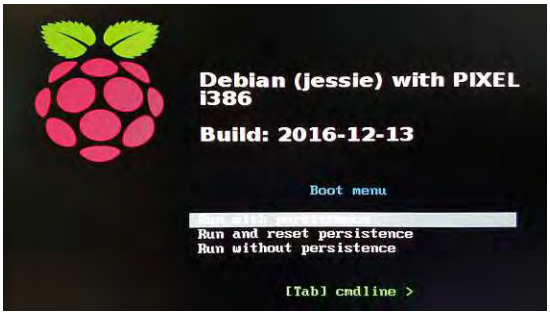
Legen Sie unsere **Heft-DVD** ins Laufwerk und starten Sie Ihren PC neu. Sollte Ihr Rechner nicht über ein DVD-Laufwerk verfügen, können Sie sich einen bootfähigen USB-Stick anlegen (siehe Seite 72). Dies ist auch die empfehlenswerte Option, wenn Sie öfter mit Debian mit Pixel arbeiten wollen. So bleiben Ihre persönliche (deutsche) Konfiguration und alle anderen Dateien permanent gespeichert.

Bei klassischen PCs – egal ob Desktop oder Notebook – spielt es keine Rolle, ob Sie sonst mit Linux oder Windows arbeiten. Die meisten PCs sind so konfiguriert, dass sie von DVD starten, sofern eine Disc beim Einschalten eigelegt ist. Falls nicht, ändern Sie die Boot-Reihenfolge (siehe Kasten rechts). Sie sollten die DVD rotieren hören und die Anzeige „Welcome to PIXEL powered by Debian“ sehen.

BOOT-MENÜ

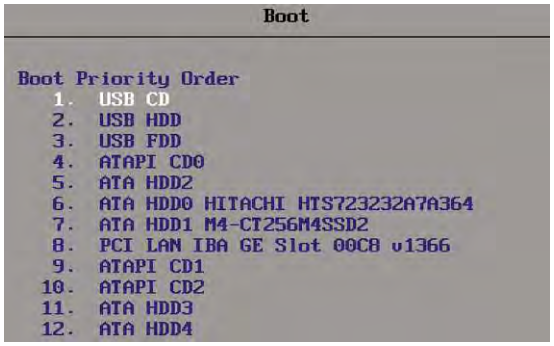
Wenn Sie während des Startvorgangs von Debian mit Pixel **[ESC]** drücken, gelangen Sie zu den Boot-Optionen. Hier gibt es drei Möglichkeiten zur Auswahl: „Run with persistence“, „Run and reset persistence“ sowie „Run without persistence“. Wenn Sie **[ESC]** nicht drücken, wird automatisch die erste Option „with persistence“, also dauerhafter Speicherung gestartet. Erwarten Sie davon allerdings nicht allzu viel: Ihre Einstellungen und Dateien sind nach einem Neustart trotzdem weg. Wenn Sie eine dauerhafte Speicherung wünschen, kommen Sie um einen USB-Stick nicht herum (siehe Seite 71).

Drücken Sie die **Tabulator**-Taste, um zu den Kommandozeilen-Optionen für den Startvorgang zu gelangen. Dies kann bei Problemen nützlich sein. So helfen Sie etwa älteren Macs beim Booten, wenn Sie **nomodeset** den Eintrag **686-pae** anfügen.



BOOT-REIHENFOLGE

Sollte Ihr PC beim Einschalten nicht von unserer Heft-DVD starten, sollten Sie die Boot-Reihenfolge ändern. Hierzu müssen Sie die grundlegenden Hardware-Einstellungen Ihres Rechners anpassen. Dies geschieht entweder über das BIOS (Basic Input/Output System) oder dessen Nachfolger, UEFI (Unified Extensible Firmware Interface). Meist gelangen Sie zu diesen Einstellungen, indem Sie beim Start **[Esc]**, **[F1]** oder **[F2]** drücken. Nutzen Sie die Pfeiltasten (BIOS) oder die Maus (UEFI), um zum Menü **Startup** oder **Boot** zu gelangen. Bringen Sie das DVD-Laufwerk an erste Stelle, speichern Sie und starten Sie den PC neu.



Falls Sie einen Mac besitzen, können Sie entweder die **[C]**-Taste gedrückt halten, wenn Sie den typischen Apple-Startsound hören („boing“), oder Sie halten **[ALT]** gedrückt und wählen die DVD als Start-Laufwerk.

Temporäre Anpassungen

Warten Sie geduldig ab, bis das System vollständig gestartet ist. Sie werden automatisch eingeloggt. Der Standard-Nutzername ist **pi**, das Passwort lautet **raspberry**. Sie können Debian mit Pixel auf Ihrem

> „Sie können auf Ihrem Rechner arbeiten wie auf einem Raspberry Pi“

Computer so verwenden wie bei einem Raspberry Pi. Stellen Sie zunächst die Tastatur auf Deutsch um (im Menü über **Preferences | Mouse and Keyboard Settings | Keyboard | Keyboard Layout | German**). Weitergehende Anpassungen lohnen sich an dieser Stelle noch nicht, denn alles, was Sie konfigurieren und alle Dateien, die Sie anlegen, werden noch nicht dauerhaft gespeichert. Wie Sie das ändern, lesen Sie auf der nächsten Seite. Über das Menü und zweimal **Shutdown** schalten Sie das System aus.

ERSTE SCHRITTE

Jetzt geht es los mit **Debian mit Pixel** auf Ihrem Computer

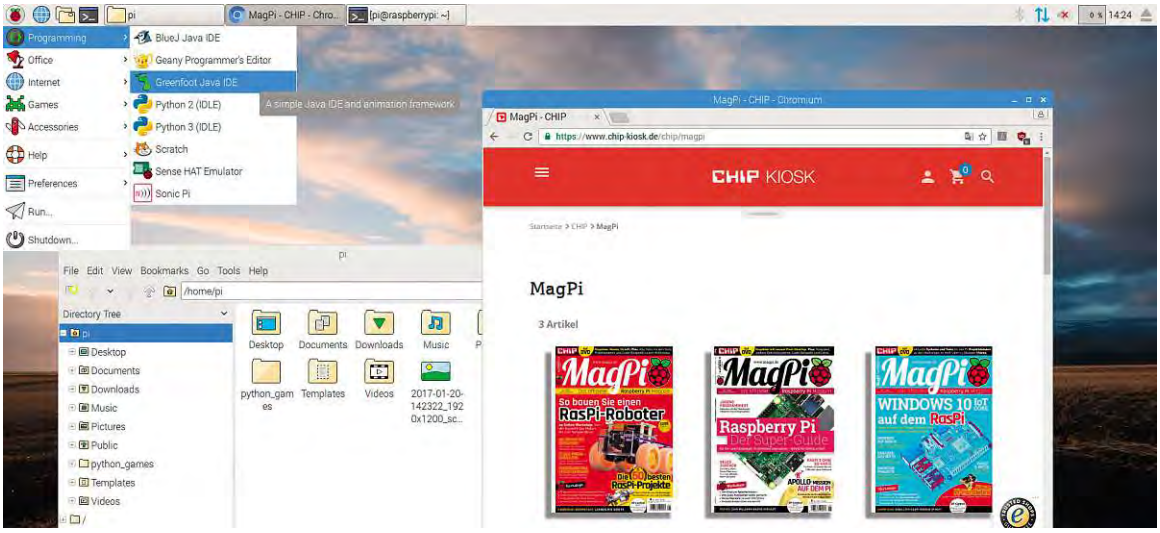
VORINSTALLIERTE PROGRAMME

Debian mit Pixel besitzt eine gute Auswahl an Software, die Sie bei Verwendung eines USB-Sticks erweitern und auf Deutsch umstellen können.

NETZWERK & INTERNET

Die WLAN-Einrichtung erfogt per Klick rechts oben auf das Netzwerksymbol. Chromium (die freie Variante von Chrome) ist vorinstalliert.

DIE OBERFLÄCHE



PIXEL-DESKTOP

Der Pixel-Desktop auf Ihrem Computer funktioniert (fast) genauso wie auf einem Raspberry Pi. Zum Lernen und Coden ist das ideal ...

Wenn Debian mit Pixel geladen ist, können Sie Ihren Computer (fast) genauso nutzen, als säßen Sie vor einem Raspberry Pi. Die vorinstallierten Programme erreichen Sie über einen Klick auf das Startmenü in der linken oberen Ecke. Hier finden Sie bereits eine gute Auswahl an Anwendungen zum Programmieren, einen Sense-HAT-Emulator sowie Lernprogramme. Zudem ist bereits Libre Office als starke Alternative zu Microsoft Office an Bord.

Allerdings sind sowohl Debian mit Pixel als auch Libre Office in der Standardkonfiguration englischsprachig. Wie Sie dies dauerhaft ändern, lesen Sie auf der nächsten Doppelseite.

Keyboard-Layout und Internet

Stellen Sie vorläufig die Tastaturbelegung auf Deutsch um (**Preferences | Mouse and Keyboard Settings | Keyboard | Keyboard Layout | German**). Wollen Sie sich drahtlos mit Ihrem Funknetz verbinden, klicken Sie einfach rechts oben auf das Netzwerk-Symbol (die zwei blauen Pfeile) und geben das Passwort ein. Hierfür ist in vielen Fällen die deutsche Tastaturbelegung unerlässlich! Falls Sie ein kabelgebundenes Netzwerk nutzen, müssen Sie gar nichts tun – Debian stellt die Netzwerkverbindung automatisch her.

Fürs Surfen im Internet ist Chromium vorinstalliert, die Open-Source-Variante von Google Chrome. Dabei ist die Flash-Unterstützung bereits aktiviert. Sie sollten also nahezu alle Online-Games und Videos ohne Probleme nutzen können.

Sie können nach Belieben Programme hinzufügen. Hierzu gibt es von Debian die APT-Paketverwaltung. Allerdings werden die Programme beim von der DVD gestarteten Live-System nur im Arbeitsspeicher zwischengespeichert und sind nach einem Neustart weg.

Das Gleiche gilt für alle Dateien, die Sie anlegen oder Einstellungen, die Sie vornehmen. Wenn Sie den Computer ausschalten, ist also alles weg. Dies kann auch Vorteile haben, denn ohne dauerhafte Speicherung haben Sie immer ein frisch aufgesetztes – allerdings englischsprachiges – System ohne Fehler.

Stick als Dauer-Speicher

Ganz klar: Eine DVD-ROM kann nicht beschrieben werden. Dementsprechend benötigen Sie einen USB-Speicherstick, der quasi im Tandem mit unserer Heft-DVD zusammenarbeitet. Dateien, die Sie anlegen, lassen sich darauf dann dauerhaft speichern – nicht jedoch Konfigurationsänderungen. So können Sie etwa mit Ihren Code-Projekten weiterarbeiten.

Stecken Sie also einen USB-Stick in Ihren Rechner und machen Sie diesen zu einem dauerhaften Speicher-Laufwerk. Das geht ganz einfach: Zunächst müssen Sie den Stick für Linux formatieren – typischerweise mit ext4. Am einfachsten geht das, wenn Sie dafür GParted installieren: Öffnen Sie ein Terminalfenster und geben Sie **`sudo apt install gparted`** ein. Anschließend starten Sie das Programm mit Administratorrechten mittels **`sudo gparted`**.

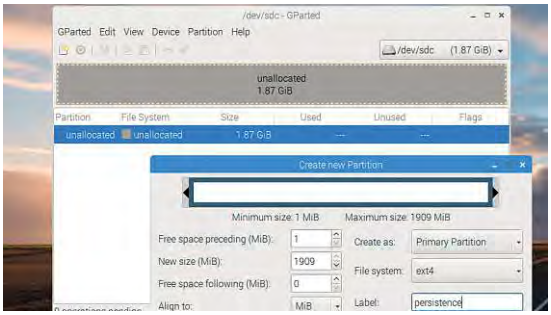
GParted besitzt eine komfortable grafische Benutzeroberfläche. Wählen Sie oben rechts Ihren USB-Stick als Laufwerk aus (Vorsicht: nicht etwa Ihre Festplatte!) und werfen Sie diesen mittels Rechtsklick und **Unmount** aus. Klicken Sie wiederum mit Rechts auf das Laufwerk und wählen Sie **Delete**. Mit einem Klick auf das grüne Häkchen starten Sie den Vorgang. Achtung: Alle Daten auf dem Stick werden dabei gelöscht.

Klicken Sie den jetzt als unallocated gekennzeichneten Speicher per Rechtsklick an und gehen Sie auf **Partition | New**. Wählen Sie **Primary Partition** und als Dateisystem **ext4**. Als Namen vergeben Sie bei „Label:“ **persistence**. Mittels Klick auf **Add** und wiederum dem grünen Haken starten Sie den Vorgang.

Nun müssen wir eine Konfigurationsdatei namens **persistence.conf** ins Stammverzeichnis unseres neuen Laufwerks hinzufügen. Entfernen Sie den USB-Stick und stecken ihn anschließend wieder ein. Öffnen Sie ein Terminal und wechseln Sie das Verzeichnis mit **`cd /media/pi`**. Mittels **`sudo chown pi persistence`** ändern Sie den Besitzer.

Als Nächstes wechseln Sie per **`cd persistence`** ins Stammverzeichnis des Sticks. Geben Sie **`sudo echo / union > persistence.conf`** ein. Damit legen Sie die Konfigurationsdatei an und schreiben dort „/ union“ hinein. Starten Sie den Rechner neu. Ihr Computer bootet nun von der DVD, nutzt aber den USB-Stick als Speicher für Dateien. Wie Sie das komplette System auf Stick speichern, lesen Sie auf der nächsten Seite.

PERSISTENCE DRIVE



Installieren Sie GParted, um den USB-Stick zu formatieren und ein dauerhaftes Laufwerk anzulegen. Auf diesem können Sie Daten speichern.

DIE GESCHICHTE HINTER DEBIAN MIT PIXEL

Wir trafen uns mit Eben Upton, dem Mitgründer der Raspberry Pi Foundation und Geschäftsführer der Raspberry Pi Trading Gesellschaft, um mit ihm die Gründe für Debian mit Pixel für PCs zu besprechen. „Ich glaube, die eigentliche Frage lautet: Warum haben wir das nicht schon viel früher gemacht?“ sagt Eben. „Viele Leute haben uns gefragt: ‚Warum macht ihr keine PC-Software? Jeder hat doch einen PC‘. Unsere Antwort war dann immer: ‚Nein, stimmt nicht. Nicht jeder hat einen PC.‘ Zudem haben viele Angst, an ihrem Rechner irgend etwas „kaputt“ zu machen.“

Daher hat sich die Raspberry Pi Foundation lange Zeit darauf konzentriert, Hardware zu entwickeln – und natürlich Software für den Raspberry Pi. „Aber dann kam Simon [Long] zu uns ins Team“, erzählt Eben. Simon hat sehr viel Arbeit investiert, um aus dem bekannten Raspbian-Desktop die deutlich aufpolierte Version mit der neuen Pixel-Oberfläche zu machen. „Eine Sache, die er wirklich großartig hinbekommen hat, war die Einbindung von Software“, erklärt Eben. Programme wie Flash, Chromium, Java und Libre Office erforderten einen hohen Integrationsaufwand. „Es kommt jetzt unserem Ziel näher, eine perfekte Umgebung zum Arbeiten und zum Coden-Lernen zu werden. Wir haben nun ein interessantes Gesamtpaket und zwingen die Nutzer auf diese Weise nicht mehr, sich einen Raspberry Pi zu kaufen, um das System einmal auszuprobieren. Jetzt schauen wir mal, ob es die Nutzer mögen“, fasst Eben zusammen.

SO GEHTS:
USB-STICK MIT PIXEL

SIE BRAUCHEN

- USB-Stick
- Auf DVD: Debian mit Pixel als ISO-Datei

Richtig komfortabel wird das Arbeiten für deutschsprachige Nutzer erst mit Pixel auf dem USB-Stick. Wir zeigen, wie das geht

Okay, seien wir ehrlich: Man kann zwar mit unserer Live-DVD sofort loslegen, was ein Riesen-Vorteil ist. Doch will man Dateien dauerhaft speichern, ist der auf der vorhergehenden Doppelseite gezeigte „Trick“ mit dem USB-Stick nur eine Hilfslösung. Deutschsprachige Anwender müssten immer noch nach jedem Neustart das Tastaturlayout umstellen. Das Betriebssystem selbst sowie die Anwendungen bleiben jedoch Englisch.

Wäre es da nicht cleverer, das ganze Betriebssystem auf den USB-Stick zu spielen, wenn man schon mit einem solchen Speicherstick arbeitet? Die eindeutige Antwort lautet: ja! Dies ist zudem bei allen Rechnern erforderlich, die gar nicht mehr über ein DVD-Laufwerk verfügen. Bei vielen schlanken Notebooks ist das beispielsweise der Fall. Zudem booten Sie per Stick deutlich schneller als von der DVD.

Um Ihnen die Arbeit zu erleichtern, haben wir Debian mit Pixel nicht nur als Live-System auf die Heft-DVD gepackt, sondern auch als Image-Datei (ISO). Mit Tools wie Etcher oder Win32DiskImager (ebenfalls auf Heft-DVD) kopieren Sie dieses System auf den Stick. Unter Mac OS X verwenden Sie stattdessen das Disk Utility, unter Linux den Befehl **dd**. Die einzige Hürde unter Linux ist es, die richtige ID für den USB-Stick herauszubekommen. Der Befehl **`lsblk`** listet alle Laufwerke auf. Mit dem Befehl **`sudo dd if=/pfadangabe/2016-12-13-pixel-x86-jessie.iso of=/dev/sdx`** (dabei die ID des Sticks beachten) beschreibt man den USB-Stick.

So kommt Pixel auf den Stick

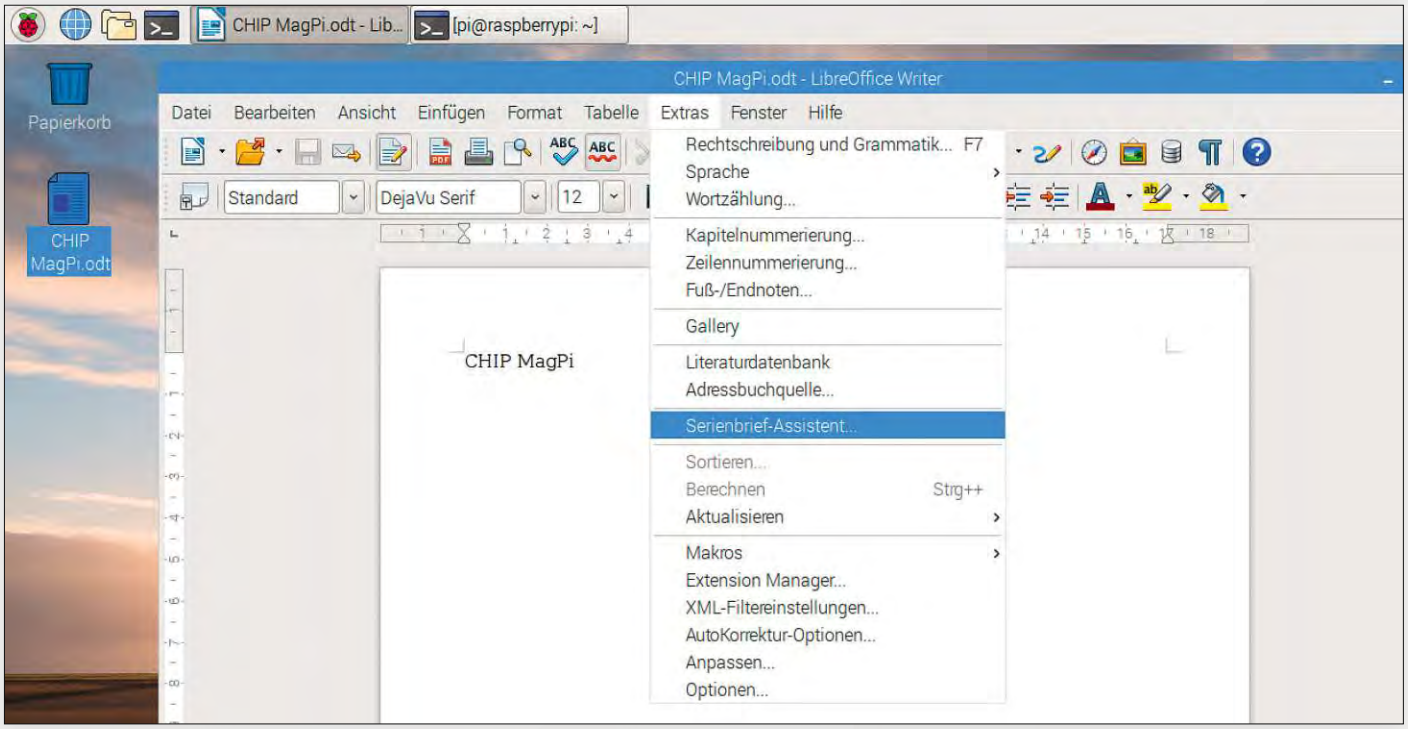
Wir zeigen an dieser Stelle, wie Sie den USB-Stick mit Etcher bespielen. Laden Sie das Tool von unserer Heft-DVD oder von **etcher.io** und installieren Sie es. Eine ausführliche Anleitung finden Sie in MagPi 06/2016 ab Seite 64. Dort wird auch gezeigt, wie Sie eine bootfähige SD-Karte für Ihren RasPi erstellen.

Starten Sie also Etcher und klicken Sie dann auf **Select Image**. Wählen Sie die ISO Datei mit dem Betriebssystem – auf unserer Heft-DVD ist es 2016-12-13-pixel-x86-jessie.iso. Sobald Sie den USB-Stick einstecken, sollte dieser automatisch unter „Select Drive“ erscheinen. Andernfalls klicken Sie auf **Select Drive** und wählen das korrekte Laufwerk. Mit einem Klick auf **Flash!** kopieren Sie das Image auf das gerade neu hinzugekommene USB-Laufwerk.

Starten Sie nun Ihren Computer. Haben Sie bisher von der DVD gebootet, dürfen Sie nicht vergessen, diese zuvor aus dem Laufwerk zu entfernen. Zudem müssen Sie jetzt erneut die Bootreihenfolge ändern (siehe Seite 69). Drücken Sie also nach dem Einschalten je nach Modell entweder [Esc], [F1] oder [F2], um ins BIOS oder UEFI zu gelangen und wählen Sie den

CHROMIUM

Als Browser ist Chromium, die Open-Source-Variante von Google Chrome, vorinstalliert. Doch auch wenn Sie Debian auf Deutsch umgestellt haben, bleiben die Menüs Englisch. Fügen Sie über die Paketverwaltung die Sprachpakete **chromium-l10n** hinzu. Chromium nutzt nun die Systemsprache.



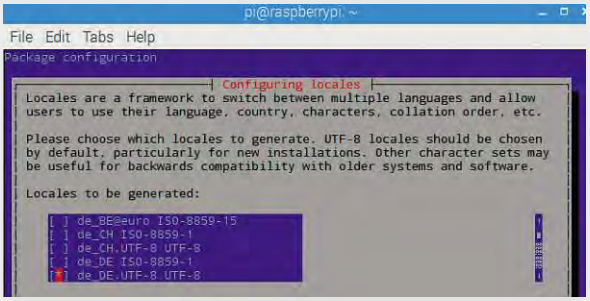
USB-Stick als Startlaufwerk. Debian mit Pixel startet nun – genauso wie von der DVD. Allerdings sollte dies jetzt viel schneller geschehen, da ein Flashspeicher deutlich schneller ist als ein optisches Laufwerk.

Unser Tipp: Verwenden Sie einen USB-3.0-Stick an einer entsprechenden USB-Schnittstelle am Rechner für maximale Performance.

Auf Deutsch umstellen

Nach dem ersten Start von Debian mit Pixel gilt auch hier wieder: Stellen Sie das Tastaturlayout auf Deutsch um. Dies geschieht über **Preferences | Mouse and Keyboard Settings | Keyboard | Keyboard Layout | German**. Das Schöne ist jetzt: Das war das letzte Mal, dass Sie dies tun mussten. Denn der USB-Stick arbeitet wie eine Festplatte als dauerhafter Speicher.

Es lohnt sich also, jetzt ein paar gründlichere Änderungen vorzunehmen, um ein deutschsprachiges System zu erhalten. Zudem können Sie Programme dauerhaft hinzufügen, das System aktualisieren oder Benutzer hinzufügen. Bei allen Änderungen, für die Sie Root-Rechte benötigen, gilt: Standardmäßig wird der Nutzer **pi** mit dem Passwort **raspberry** angelegt.



Oben Eine Textverarbeitung wie Libre Office Writer macht für deutschsprachige Nutzer erst dann Sinn, wenn man die Sprache umstellt

Links Die „locales" legen die Systemsprache für Debian mit Pixel fest und lassen sich ändern

Bringen Sie das System zunächst auf den aktuellen Stand: Mit `sudo apt update` aktualisieren Sie die Paketlisten, mit `sudo apt upgrade` führen Sie das Update durch. Dies kann beim ersten Mal sehr viel Zeit in Anspruch nehmen, das ist normal.

Anschließend stellen Sie das gesamte System auf Deutsch um: Öffnen Sie die Standorteinstellungen über eine Konsole, indem Sie dort `sudo dpkgreconfigure locales` eingeben. Gehen Sie mit den Pfeiltasten zum Eintrag **de_DE.UTF-8 UTF-8** und markieren Sie ihn mit der Leertaste. Wechseln Sie mit der Tabulatortaste zu **Ok** und drücken Sie die Eingabetaste. Im folgenden Dialog können Sie diese Spracheinstellungen zum Standard machen: Wählen Sie mit den Pfeiltasten **de_DE.UTF-8**, wechseln Sie mit der Tabulatortaste zu **Ok** und drücken Sie die Eingabetaste. Starten Sie den Rechner mit `sudo reboot` neu.

Wollen Sie weitere Anwendungen installieren oder deutsche Sprachpakete nachladen, empfehlen wir die Installation des grafischen Paketmanagers Synaptic. Dies erledigen Sie über `sudo apt install synaptic`.

LIBRE OFFICE

Prima: Debian mit Pixel bringt standardmäßig das mächtige Libre Office mit, eine gute Alternative zu Word & Co – allerdings auf Englisch. Um das zu ändern, installieren Sie die grafische Paketverwaltung **Synaptic** (siehe rechts unten) und installieren Sie darüber das deutsche Sprachpaket **libreoffice-l10n-de**. Starten Sie den Libre Office Writer und wählen Sie unter **Tools | Options | Language Settings | Languages** überall **German** aus.

PHIL KING
Wenn er nicht gerade an der *MagPi* arbeitet, plant Phil neue Pi-Projekte und programmiert mit seinem Sohn in Scratch.
@philking68

SOUND AUF KNOPFDRUCK: MUSIKBOX

Spielen Sie mit zwei oder mehr Knöpfen verschiedene Sounds ab

Sie brauchen

- GPIO Zero
- 1× lötfreie Platine
- 2× Schaltknöpfe
- 3× Male/Female-Jumperkabel
- 2× Male/Male-Jumperkabel
- Lautsprecher oder Kopfhörer

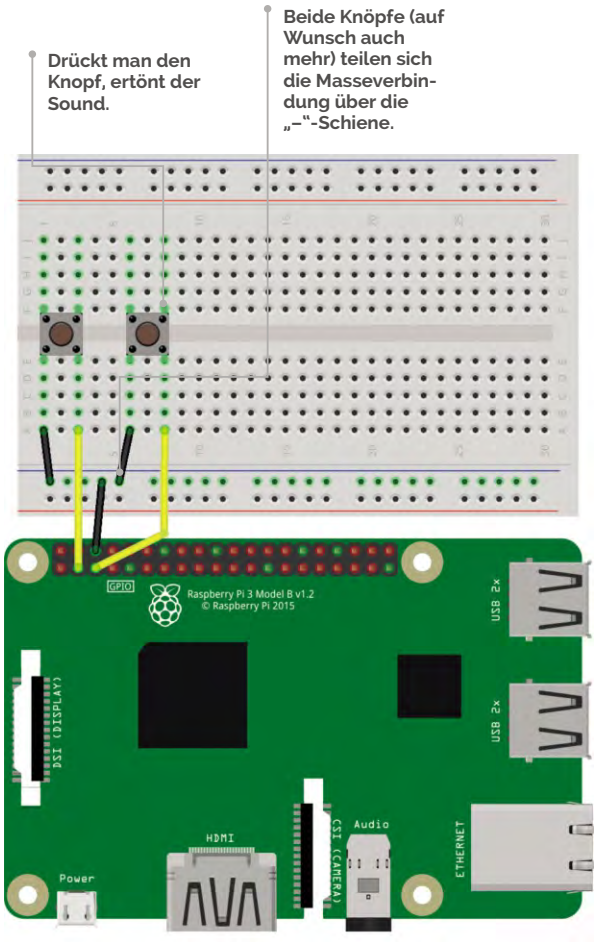
In diesem Projekt bauen wir eine GPIO-gesteuerte Musikbox mit zwei Schaltknöpfen. Drücken wir sie, ertönen verschiedene Sounds.

>SCHRITT 01

Klänge sammeln

Im Terminal legen wir einen neuen Ordner an: **`mkdir musicbox`**. Hereinspaziert mit **`cd musicbox`**. Jetzt brauchen wir Sounds als Quelle. Wir benutzen einige in Scratch eingebaute Percussion-Klänge, die bereits auf dem Raspberry Pi zur Verfügung stehen. Im Terminal erstellen wir mit **`mkdir samples`** den Zielordner und wechseln hinein: **`cd samples`**. Als Nächstes kopieren wir die gewünschten Sounds wie folgt:

```
cp /usr/share/scratch/Media/Sounds/
Percussion/* .
```



>SCHRITT 02

Trommel schlagen

Zuerst schreiben wir ein Python-Programm, das ein Drumsample in einer Schleife abspielt. In IDLE, der Entwicklungsumgebung von Python 3, erzeugen wir eine neue Datei und fügen den Code aus **ch4listing1.py** ein. Wir speichern in den Ordner **musicbox**. Mit **F5** spielt das Programm den Sound ab. Hören wir nichts, aktivieren wir die Audioschnittstelle im Terminal mit **`amixer cset numid=3 1`** (analoge Ausgabe) oder **`amixer cset numid=3 2`** (HDMI-Ausgabe).

>SCHRITT 03

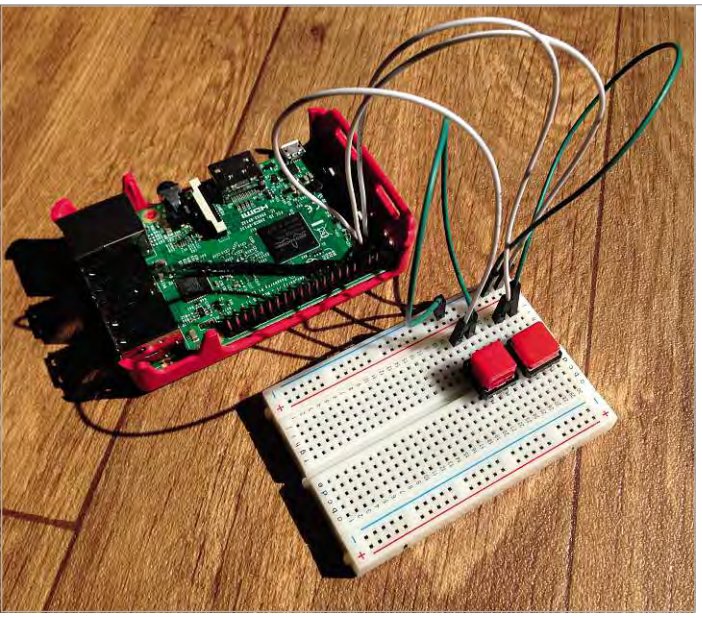
Knöpfe verdrahten

Jetzt wird der Pi ausgeschaltet, um die Schaltung einzurichten. Den Knopf platzieren wir so aufs Steckbrett, dass er die Riefe in der Mitte überbrückt. Ein Kontakt ist mit dem GPIO-Pin 2 verbunden, der andere wird mit dem Masse-Verteiler des Steckboards verbunden, der wiederum mit einem Masse-Pin des Pi verbunden wird.

Der Sound soll ertönen, wenn der Knopf gedrückt wird. In IDLE öffnen wir eine neue Datei und fügen den Code aus **ch4listing2.py** ein. Wieder speichern wir in **musicbox**. Zum Start importieren wir die **`Button`**-Klasse aus GPIO Zero und die **`pause`**-Klasse aus der **`signal`**-Library. Mit **`button = Button (2)`** weisen wir die Knopf-Variable GPIO-Pin 2 zu. Den Sound weisen wir mit **`button.when_pressed = drum.play`** zu, wenn der Knopf gedrückt wird. Zum Schluss legt **`pause()`** fest, dass das Programm wartet, bis der Knopf erneut gedrückt wird. Wenn es läuft, sollte per Knopfdruck der zugeordnete Sound zu hören sein.

Sprache
>PYTHON 3

DOWNLOAD:
magpi.cc/2bhwqlH



Oben Immer wenn einer der Knöpfe gedrückt wird, ertönt der ihm zugeordnete Klang über die angeschlossenen Lautsprecher

>SCHRITT 04

Knopf zwei

Wie im Diagramm gezeigt, setzen wir den zweiten Knopf auf und verdrahten ihn mit GPIO 3 und Masse. Erneut legen wir eine Datei mit IDLE an, geben den Code von **ch4listing3.py** ein und speichern in **musicbox**. Wir weisen die **Button**-Objekte und Sounds den Pins nicht individuell zu. Hierzu nutzen wir die Dictionary Structure, eine pythonspezifische Datenstruktur:

```
sound_pins = {
    2: Sound("samples/DrumBizz.wav"),
    3: Sound("samples/CymbalCrash.wav"),
}
```

Danach erzeugen wir eine Liste der Knöpfe auf allen Pin-Nummern im **sound_pins**-Dictionary: **buttons = [Button(pin) for pin in sound_pins]**

Zum Schluss schreiben wir eine **for**-Schleife, die für jeden Knopf im Dictionary nachschlägt und den entsprechenden Sound abspielt:

```
for button in buttons:
    sound = sound_pins[button.pin.number]
    button.when_pressed = sound.play
```

Läuft das Programm, sollten über beide Knöpfe verschiedene Sounds zu hören sein.

>SCHRITT 05

Noch mehr Schalter

Auf diese Weise lassen sich ganz leicht weitere Knöpfe aufs Steckboard stecken und zum Klingen bringen. Sie werden mit GPIO-Pins (ausschließlich den nummerierten) und wie zuvor mit dem Masse-Verteiler verbunden. Danach trägt man die GPIO-Pin-Nummern und Sounds ins Dictionary wie folgt ein:

ch4listing1.py

```
import pygame.mixer
from pygame.mixer import Sound

pygame.mixer.init()

drum = Sound("samples/DrumBuzz.wav")

while True:
    drum.play()
```

ch4listing2.py

```
from gpiozero import Button
import pygame.mixer
from pygame.mixer import Sound
from signal import pause

pygame.mixer.init()
button = Button(2)
drum = Sound("samples/DrumBuzz.wav")

button.when_pressed = drum.play
pause()
```

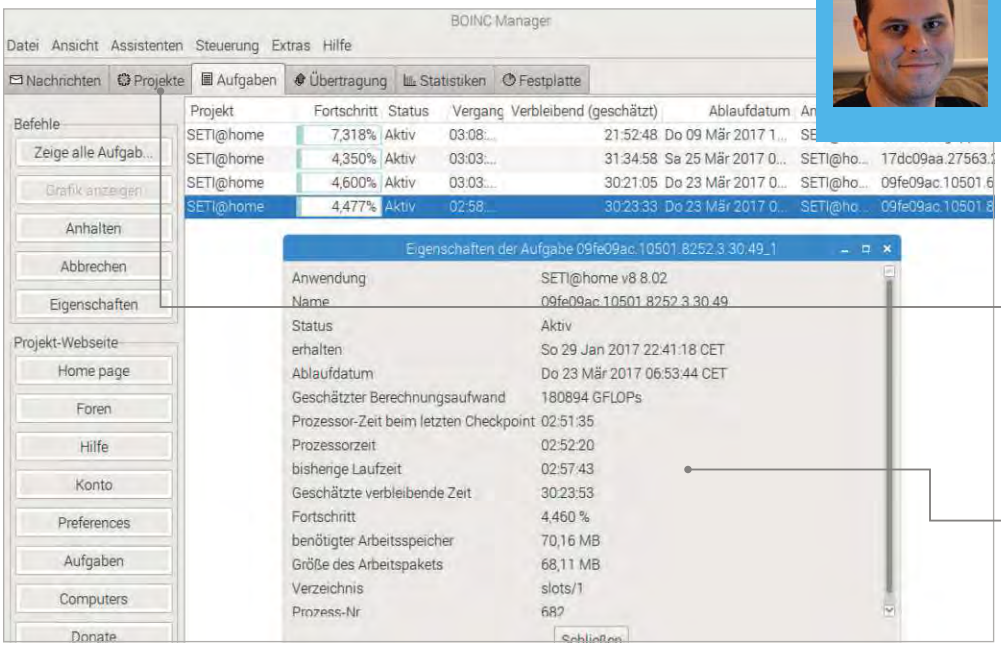
ch4listing3.py

```
from gpiozero import Button
import pygame.mixer
from pygame.mixer import Sound
from signal import pause

pygame.mixer.init()

sound_pins = {
    2: Sound("samples/DrumBuzz.wav"),
    3: Sound("samples/CymbalCrash.wav"),
}

buttons = [Button(pin) for pin in sound_pins]
for button in buttons:
    sound = sound_pins[button.pin.number]
    button.when_pressed = sound.play

pause()
```

```
sound_pins = {
    2: Sound("samples/DrumBizz.wav"),
    3: Sound("samples/CymbalCrash.wav"),
    4: Sound("samples/Gong.wav"),
    14: Sound("samples/HandClap.wav"),
}
```

ROB ZWETSLOOT

Tüftler und Bastler, begeisterter Cosplayer, kümmert sich um Themenschwerpunkte bei der MagPi.
magpi.cc / @TheMagP1



Legen Sie mit dem Reiter „Projekte" fest, welchen wissenschaftlichen Vorhaben Sie Rechenkapazitäten spendieren wollen. In diesem Beispiel ist es „SETI@home"

Der BOINC-Manager informiert Sie detailliert, an welchen Aufgaben der Raspberry Pi gerade arbeitet und wie viele Ressourcen dafür aktuell verbraucht werden

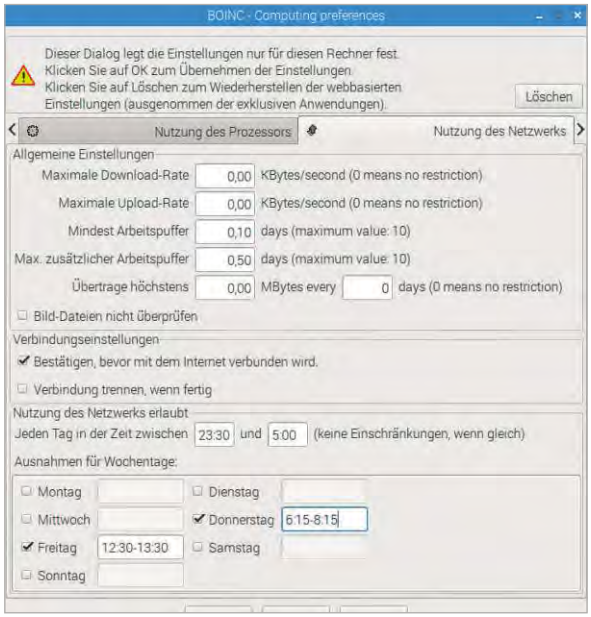
FORSCHUNGSPROJEKTE UNTERSTÜTZEN MIT BOINC

Sie brauchen

- BOINC-Client: **boinc.berkeley.edu**
- Raspberry Pi
- Internet-Verbindung

Ihr Raspberry Pi langweilt sich? Dann spenden Sie doch etwas Rechenkapazität für wissenschaftliche Projekte

Der Raspberry Pi liegt mal wieder auf der faulen Haut, weil die Hardware nicht ausgelastet ist? Schade eigentlich, denn Rechenkapazität wäre genügend vorhanden – insbesondere beim neuesten Modell, sprich dem Raspberry Pi 3. Was also tun? Unser Vorschlag: Spenden Sie die überschüssige Rechenleistung einem der vielen Forschungsprojekte, die sich zum Beispiel dem Kampf gegen gefährliche Krankheiten, verbesserten Klimamodellen oder der Erforschung des Weltraums widmen.



Optimiert Passen Sie den Raspberry so an, dass seine Leistung optimal genutzt wird

Das Grundprinzip ist immer noch das Gleiche wie bei „SETI@home", einem Projekt, das in den Neunzigerjahren startete: Geforscht wird nach außerirdischen Lebenszeichen, wofür man die Rechenkraft von unzähligen Computern auf der ganzen Welt geschickt bündelt. Technische Grundlage für dieses Verfahren ist das vom SETI@home-Team entwickelte BOINC.

Die Software dient aber nicht nur als Plattform für verteiltes Rechnen beim SETI-Projekt. BOINC verfolgt einen universellen Ansatz und lässt sich auch für andere wissenschaftliche Vorhaben einsetzen. Aktuell wird zum Beispiel an einem genaueren 3D-Modell unserer Milchstraße gearbeitet, ebenso steht die Simulation von Proteinfaltungen für die Erforschung von neuen Medikamenten auf dem Plan.

Es steht Ihnen frei, welche Projekte Sie mit Ihrem Raspberry Pi fördern. Ein Vorteil von BOINC ist, dass die zur Verfügung stehende Rechenzeit automatisch einem anderen Projekt zugeteilt wird, wenn einem Forschungsvorhaben die Datenpakete kurzfristig ausgehen sollten. Das setzt natürlich voraus, dass Sie mindestens zwei Projekte unterstützen.

>Schritt-01

Raspbian aktualisieren

BOINC ist auch für Raspbian verfügbar, deshalb bereitet die Installation der Pakete keinerlei Probleme. Zuerst müssen Sie Raspbian auf den neuesten Stand bringen. Öffnen Sie das Terminal und geben Sie ein:

```
sudo apt update
sudo apt upgrade
```

Ist das Update erledigt, folgt der nächste Schritt, nämlich die Installation von BOINC:

```
sudo apt install boinc
```

Dieser Befehl richtet BOINC sowie den BOINC-Manager ein, mit dem Sie Ihre Projekte wählen. Zusätzlich liefert das Tool aussagekräftige Statistiken.

>Schritt-02

Software starten

BOINC arbeitet sowohl im Kommandozeilenmodus als auch mit einer grafischen Benutzeroberfläche. Falls Sie die Kommandozeile bevorzugen – kein Problem, aber diese Art der Bedienung setzt etwas Einarbeitungszeit voraus, siehe: **magpi.cc/2hgkTYO**.

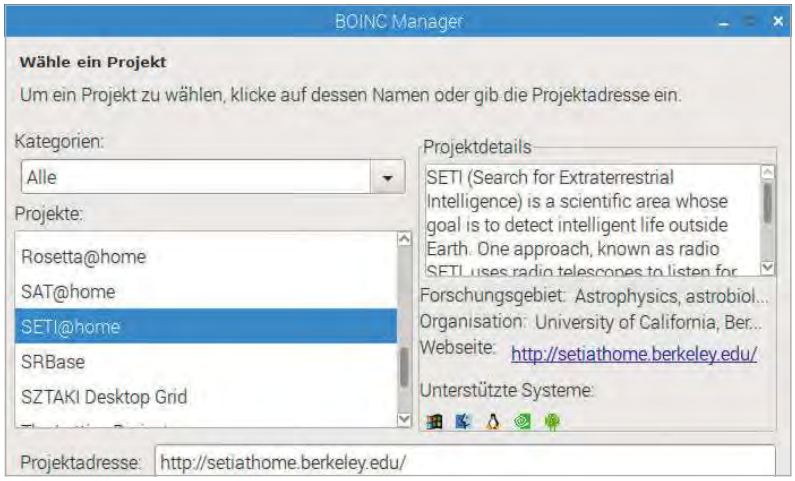
In diesem Tutorial konzentrieren wir uns deshalb auf die grafische Benutzeroberfläche des Tools. Sie finden die Software im Desktop-Menü des Pixel-Desktops. Starten Sie das Programm (siehe **Systemwerkzeuge | BOINC Manager**).

>Schritt-03

Wählen Sie ein Projekt

Der BOINC-Manager zeigt Ihnen an, welche Projekte aktuell zur Verfügung stehen. Wählen Sie das Projekt, in diesem Beispiel ist es „SETI@home“. Klicken Sie auf **Weiter**. Nun dauert es eine Weile, bis sich die Verbindung aufbaut. Ist der Kontakt hergestellt, blendet der BOINC-Manager ein Abfragefenster ein.

Da dies vermutlich Ihr erster Besuch ist, haben Sie noch keine Kontaktdaten, die Sie im Formular eintragen könnten. Geben Sie deshalb Ihre E-Mail-Adresse und ein Passwort an. Bestätigen Sie mit **Weiter** und schließen Sie die Konfiguration mit **Beenden** ab.



Projekte Suchen Sie sich aus der obigen Liste Ihr favorisiertes Projekt heraus

>Schritt-04

BOINC konfigurieren

Im Prinzip können Sie die Werkseinstellungen von BOINC so lassen, wie sie sind. Aber vielleicht wollen Sie präziser vorgeben, wie viel Rechenleistung Sie für wissenschaftliche Projekte abzweigen wollen oder Sie möchten verhindern, dass externe Speichermedien (etwa eine Festplatte) zu stark beansprucht werden. Solche Limits legen Sie im BOINC-Manager unter **Assistenten | Einstellungen** fest. Vorher sollten Sie aber mit **Ansicht | Erweiterte Ansicht** auch die verborgenen Optionen des Konfigurationsmanagers aktivieren. Dann haben Sie auch Zugriff auf die Netzwerkeinstellungen und können die Auslastung des RasPis auf einzelne Tage beschränken.

>Schritt-05

BOINC nutzen

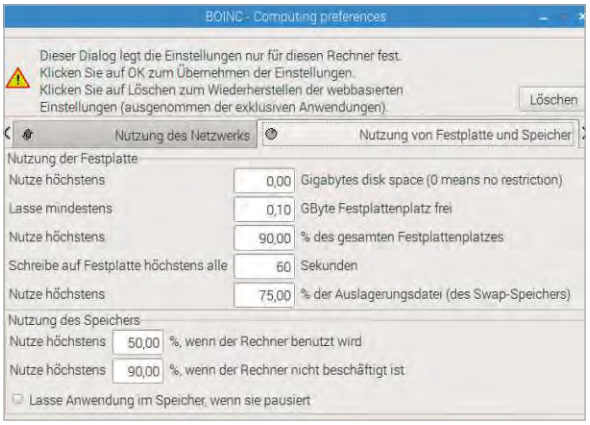
Haben Sie Ihre Projekte ausgewählt, lädt BOINC ab sofort selbstständig Datenpakete herunter und kümmert sich darum, dass diese bearbeitet werden. Ihr Raspberry Pi ist nun ein Teil der weltumspannenden wissenschaftlichen Community!

STROMVERBRAUCH

Die Projekte sind rechenintensiv und verbrauchen zusätzliche Energie. Deshalb lohnt es sich, die Einstellungen von BOINC zu kontrollieren.

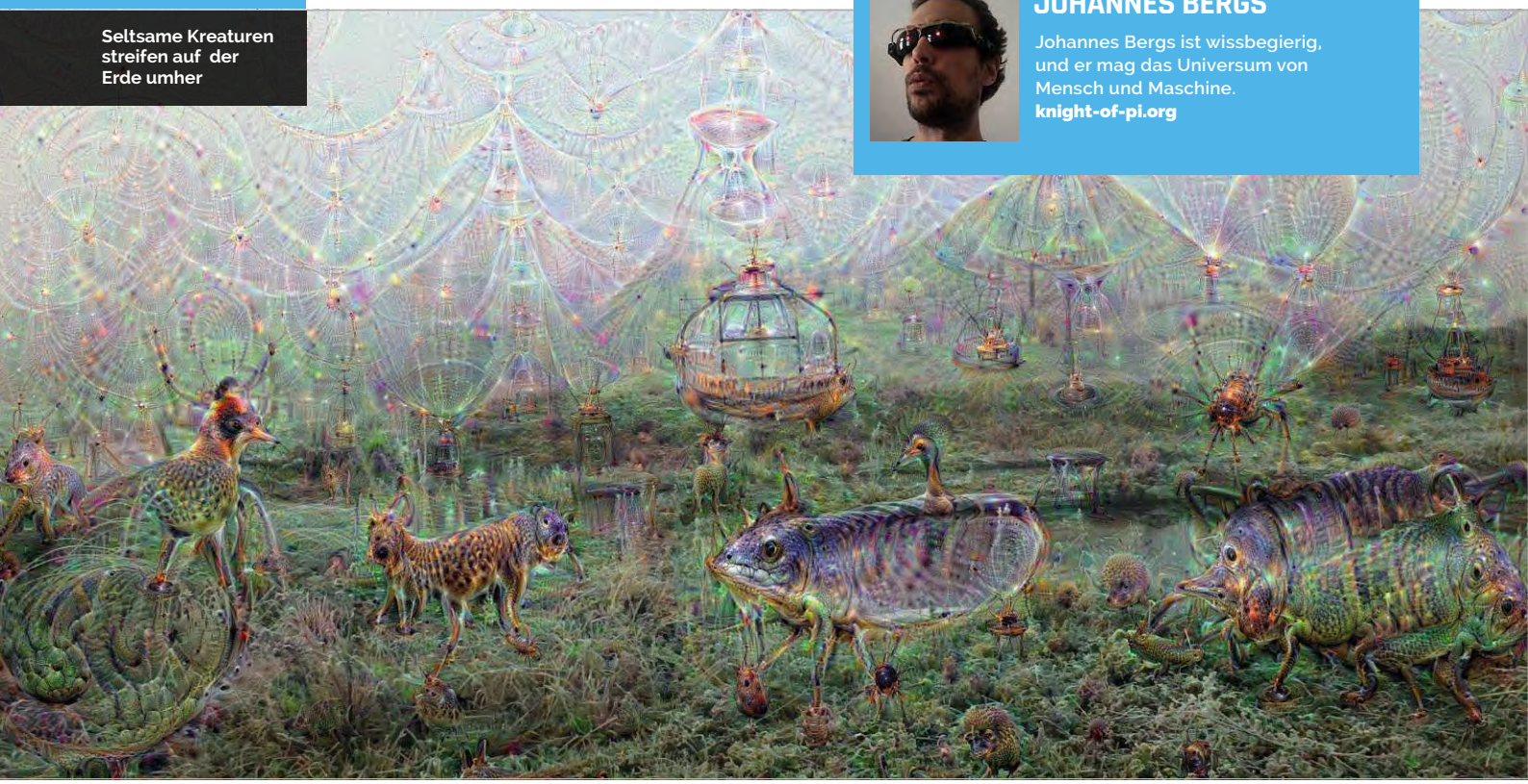
TEAMARBEIT

Bei den meisten Projekten können Sie sich mit anderen Usern in Teams organisieren oder Freundeslisten pflegen, siehe Screenshot rechts.



Ressourcen Sie können den Zugriff auf den Hauptspeicher oder die Festplatte individuell regeln oder ganz einschränken

Account Mit BOINC greifen Sie direkt auf die jeweilige Homepage Ihres Projekts und den dazugehörigen Account zu – sehr praktisch



Seltsame Kreaturen streifen auf der Erde umher

JOHANNES BERGS

Johannes Bergs ist wissbegierig, und er mag das Universum von Mensch und Maschine.
knight-of-pi.org

GOOGLE DEEPDREAM AUF DEM RASPI

Sie brauchen

- Kamera-Modul **magpi.cc/28ljlsz**
- PsyCam **magpi.cc/2eCSQOD**

WELCHEN RASPI VERWENDEN?

Aufgrund der höheren Rechenleistung ist ein Pi 3 für das Projekt empfehlenswert. Aber auch ein Pi 2 kommt in Frage.

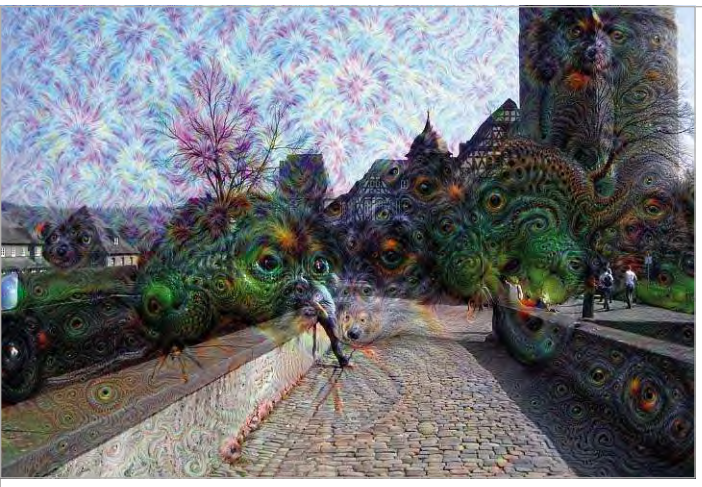
Träumen Computer von elektrischen Schafen? Mit dem Pi und Googles DeepDream-Algorithmus erzeugen Sie surreale Traumbilder

Im Traum produziert unser Gehirn mal Monster, mal wunderbar merkwürdige Welten und Kreaturen. Was aber sähe ein träumender Computer? Installiert man Googles DeepDream auf dem Raspberry Pi, bekommt man einen Eindruck davon. DeepDream ist das Nebenprodukt eines neuronalen Netzwerks zur Objekterkennung. Die Software wurde so verändert, dass sie jedes Ding, das sie erkennt, auf ihre sehr eigene Weise darstellt. Die so erzeugten Bilder sind absolut surreal.

Neuronale Netzwerke lassen sich auf eine ganze Reihe von Aufgaben anwenden, etwa Gesichtserkennung, Übersetzungen oder zum Spielen von Go. Caffe, das Framework des neuronalen Netzwerks, das DeepDream zugrunde liegt, wurde ursprünglich zur Identifizierung von Objekten in Bildern entwickelt. Das Programm selbst hat man mit einer gigantischen Menge an Bildern gefüttert, um die wichtigsten Eigenschaften eines Untersuchungsobjekts erkennen zu können.

Künstliche Neuronen ahmen ihre natürlichen Kollegen aus dem Gehirn nach. Jedes einzelne besitzt in DeepDream einen Schwellenwert, an dem es aktiv wird. Außerdem hat es zahlreiche interpretierende ein- und ausgehende Verbindungen zu anderen Neuronen. Wenn die Summe der jeweils gewichteten eingehenden Verbindungen die Aktivierungsschwelle überschreitet, sendet das Neuron ein Signal.

Ein neuronales Netz besteht aus mindestens drei Neuronen-Schichten. Der Input-Layer nimmt das Signal auf und reicht es an versteckte Schichten weiter. Diese verarbeiten die Information. Die letzte Schicht ist für die Ausgabe des Signals zuständig. Der Eingangslayer eines neuronalen Netzwerks, das etwa die Ziffern „0“ bis „9“ in Bildern mit einer Größe von 28 mal 28 Pixels erkennen soll, hat 784 Neuronen: eins für jedes Pixel. Bekommt das künstliche Hirn ein Bild, produziert jedes Neuron in der ersten Schicht ein Signal, dessen Stärke dem Grauwert des korrespondierenden Pixels entspricht: je dunkler, desto stärker. Jedes Eingansneuron ist mit allen anderen Neuronen des ersten versteckten Layers verbunden. Und jede Verbindung hat eine definierte Priorität. Je höher



Oben Magische Stadt in Fantasia

diese ist, umso mehr vom Signal des Eingangsneurons erreicht das versteckte Neuron. Überschreitet die Eingangsstärke die Aktivierungsschwelle des versteckten Neurons, feuert es ein Signal an alle Neuronen des nächsten Layers. Die Signalintensität ist also abhängig vom Grad der Priorisierung der Verbindung und der eingehenden Signalstärke. Wenn die Aktivierungsschwelle eines Ausgabe-Neurons von einer priorisierten Eingabe-Verbindung überschritten wird, ist die daraus resultierende Signalstärke das Maß der Vertrauenswürdigkeit der Klassifizierung.

Prioritäten und Schwellenwerte werden zufällig initialisiert. Das produziert unpräzise Ergebnisse: Ein Bild aus dem Trainingspool zeigt etwa eine „5“, aber das Ausgabe-Neuron von „2“ erzeugt ein starkes Signal. Durch das kontinuierliche Hin- und Herschieben der Werte zwischen den Neuronen auf den Layern wird ein Fehler schrittweise beseitigt. Dieser Prozess wird für jede Bilddatei im Trainingspool so lange wiederholt, bis sich die Resultate verbessern.

DeepDream in der Praxis

Statt akkurater Objekterkennung modifiziert DeepDream im Unterschied zu anderen neuronalen Netzwerken das Eingangsbild, um es dem Bildinhalt anzugleichen, den es aus den Trainingsdaten kennt. Ein Beispiel: Das Bild einer Wolke hat Strukturen, die entfernte Ähnlichkeiten mit einem Hund zeigen. DeepDream assoziiert dann „Hund“. Aus dem Input-Bild wird durch die Gestalt von Hunden, die DeepDream aus seinem Datenbestand „kennt“, etwas Hundeähnliches errechnet.

Zusätzlich berücksichtigt die Software die Tiefe der versteckten Layer im Verlauf ihrer Auswahlprozesse. Ein Ausgabebild weist demgemäß mehr und feinere Details auf, je näher der ausgewählte versteckte Layer am Output-Level liegt. Daher entstehen entweder Bilder aus einfachen Grundformen oder mit detaillierten, traumartigen Kreaturen.

Auf **deepdreamgenerator.com** können Sie sich einen ersten Eindruck von Google DeepDream verschaffen. Wie Sie Deepdream auf dem Raspberry Pi zum Laufen bringen, erfahren Sie nachfolgend.

DeepDream auf dem Pi

Wir booten den Raspberry Pi ins Raspbian-Jessie-System. Das Kamera-Modul sowie die notwendige Peripherie sind bereits angeschlossen. Im Terminal installieren wir jetzt die DeepDream-Software:

Sprache
>PYTHON

DOWNLOAD:
magpi.cc/2eCSQOD

```
mkdir ~/deepdream && cd ~/deepdream
git clone https://github.com/JoBergs/PsyCam
cd PsyCam
python install_tools.py packages
python install_tools.py caffe
python install_tools.py protobuf
python install_tools.py camera
sudo reboot
```

Nach der GitHub-Installation spielen wir per **pip** oder **apt** alle nötigen Pakete ein. Danach installieren wir das Open-Source-Framework für neuronale Netzwerke namens Caffe. Weil Geschwindigkeit eine Rolle spielt, sollten wir zudem Googles Protobuf für serielle Daten benutzen. Vor dem Reboot wird zudem die Kamera aktiviert. Geduld ist gefragt, denn bis alles bereit ist, vergehen ein paar Stunden. Der Vorgang sollte übrigens auch auf jeder neueren Ubuntu-Installation arbeiten. Tauchen Probleme auf, kann man das System auch von Hand installieren (siehe **magpi.cc/2eCSxDt**). Den Pi lassen wir jetzt wie folgt träumen:

```
cd ~/deepdream/PsyCam
python psycam.py
```

Die Netzwerk-Parameter „depth“ (**-d**), „octave“ (**-o**) und „type“ (**-t**) sind zufallsbasiert. Das Beigeben von **-c** führt zu dauerhaften Träumen („continuously“). Mit **-i BILD.jpg** nutzen wir eine Bilddatei anstelle der Kamera. Um mehr über die Input-Argumente zu erfahren, tippt man Folgendes ins Terminal:

```
python psycam.py -help
```

Wenn der Pi aus den Träumen erwacht, finden wir die Ergebnisse inklusive Input unter **/home/pi/deepdream/PsyCam/dreams** mit einem Timestamp versehen. Die Bilder des Unbewussten unseres Raspberry Pi lassen sich jetzt mit einem Foto-Viewer bestaunen.

DEEPDREAM IM GOOGLE BLOG

Alexander Mordvintsev, Christopher Olah und Mike Tyka haben einen sehr lesenswerten (englischsprachigen) Artikel zu DeepDream verfasst: **magpi.cc/2eCRHH3**

KIVY GUI

Das mächtige Python-GUI-Framework Kivy ist raffiniert genug, um DeepDream eine ansprechende Gestalt zu geben.

Alice im sehr bunten Wunderland

ROB ZWETSLOOT

Tüftler und Bastler, begeisterter Cosplayer, kümmert sich um Themenschwerpunkte bei der MagPi.
magpi.cc / @TheMagP1

PISPY-CAM

Schützen Sie sich vor unliebsamen Zeitgenossen: Installieren Sie eine Überwachungskamera mit Bewegungssensor

Sie brauchen

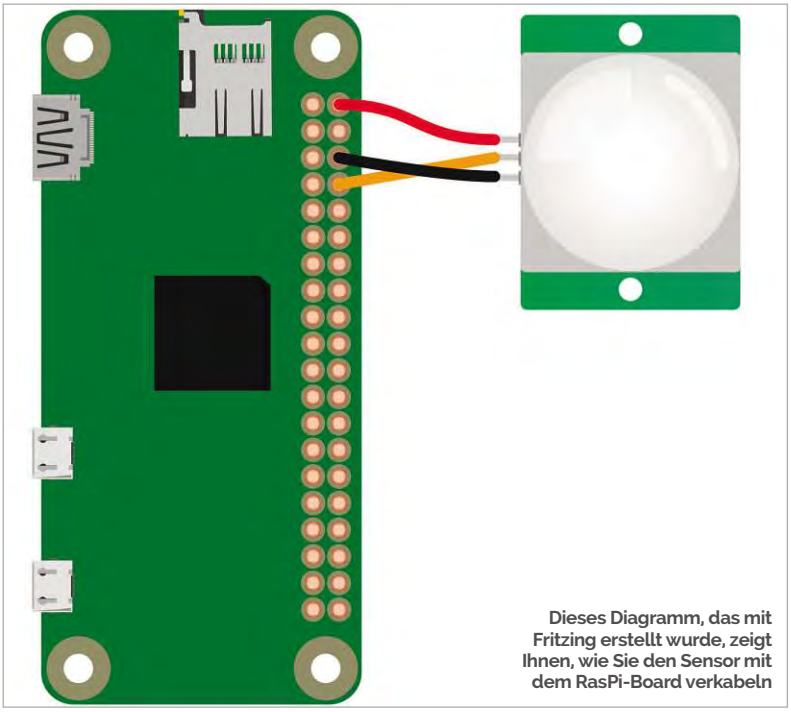
- Pi Zero
- PIR Bewegungssensor **bit.ly/2kEuutN**
- Raspberry Pi Kamera-Modul **magpi.cc/28ljlsz**
- Pi Zero Kamerakabeladapter **bit.ly/2ki4SSl**
- Batterie

Vergessen Sie die teure Überwachungstechnik aus dem Versandhandel. Es geht viel preiswerter: Alles was Sie benötigen, ist ein Pi Zero Version 1.3, ein Sensor und ein Kameramodul.

>Schritt-01

Kamera anschließen

Es ist leicht, eine Kamera am Pi Zero anzuschließen. Allerdings benötigen Sie dazu ein spezielles Adapterkabel, da der Kamera-Port des Pi Zero etwas schmaler ist – der kompakteren Platine wegen. Wenn Sie das Adapterkabel tauschen, achten Sie bitte darauf, dass Sie die filigranen Anschlüsse des Kameramoduls nicht beschädigen. Orientieren Sie sich bei der Montage am Original-Flachbandkabel. Die schmale Seite des Adapterkabels stecken Sie anschließend in die Kamera-Anschlussbuchse des Pi Zero. Das „Camera Serial Interface" (CSI) unterscheidet übrigens das Pi Zero 1.3 vom Vorgängermodell – dort fehlt es.



Dieses Diagramm, das mit Fritzing erstellt wurde, zeigt Ihnen, wie Sie den Sensor mit dem RasPi-Board verkabeln

>Schritt-02

Sensor anschließen

Ein Vorteil der folgenden Schaltung ist, dass man keine Widerstände für den Bewegungssensor (PIR) benötigt. Der Sensor hat drei Anschlüsse: VCC, GND, und OUT. VCC wird mit einem 5-Volt-Pin verbunden, GND mit einem Ground-Pin (Masse). Zu guter Letzt haben wir dann noch OUT: Diesen Anschluss des Sensors verkabeln Sie mit Pin 8, also mit GPIO 14.

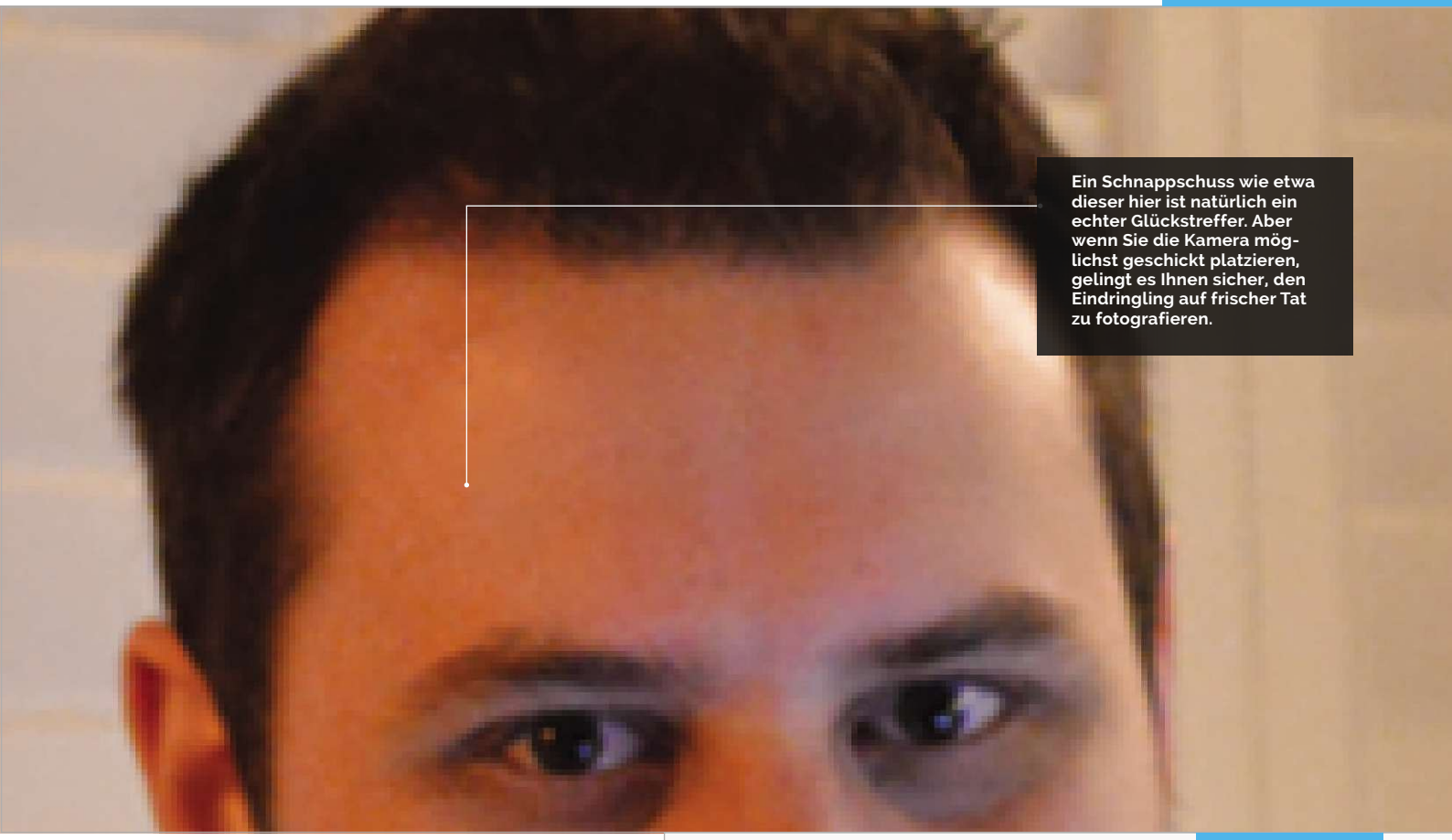
Wie Sie die Kabel am Pi Zero anschließen, hängt von der jeweiligen Hardware-Konfiguration der Platine ab. Falls eine Pin-Leiste vorhanden ist, können Sie die Drähte mit Steckern versehen und dann aufstecken. Man kann die Kabel auch direkt an die Platine anlöten – eine Steckerlösung ist aber besser, da sich der RasPi dann auch noch für andere Aufgaben einsetzen lässt.

>Schritt-03

Das Programm einspielen

Wie immer bieten wir Ihnen die Option, den Code von unserer **Heft-DVD** herunterzuladen – das spart Zeit und reduziert Fehlerquellen. Es steht Ihnen natürlich frei, den Code abzutippen. Der Code setzt zwei Bibliotheken voraus: zum einen „GPIO Zero", zum anderen „Picamera". GPIO Zero benötigen wir, um auf möglichst einfache Weise die Daten auszulesen, die der PIR-Bewegungssensor liefert. Dieser Output wird wiederum in den Picamera-Code eingebunden.

Ziel ist es, ein Beweisfoto aufzunehmen, sobald der Sensor eine Bewegung registriert. Parallel erfasst das Programm die Aufnahmezeit des Fotos, dann wartet



Ein Schnappschuss wie etwa dieser hier ist natürlich ein echter Glückstreffer. Aber wenn Sie die Kamera möglichst geschickt platzieren, gelingt es Ihnen sicher, den Eindringling auf frischer Tat zu fotografieren.

es fünf Sekunden, bevor das nächste Bild aufgenommen wird. Speichern Sie den Code unter der Dateibezeichnung **spy.py** im Home-Verzeichnis.

>Schritt-04

Ein erster Testlauf

Die Empfindlichkeit des PIR-Sensors lässt sich regeln. Dazu dienen die orangefarbenen Schrauben auf der Platine. Sorgen Sie als Nächstes dafür, dass die Daten unmittelbar im Log gespeichert werden. Starten Sie das Terminal und editieren Sie die Konfigurationsdatei

> Verstecken Sie die Kamera so, dass sie nicht gesehen wird

„profile“. Dazu dient der folgende Befehl **sudo nano /etc/profile**. Fügen Sie am Ende dieser Datei die folgende Zeile ein: **sudo python spy.py**

>Schritt-05

Abschließende Arbeiten

Die Überwachungskamera sollte natürlich nicht gefunden beziehungsweise gesehen werden. Also brauchen Sie einen guten Platz, um sie zu verstecken. Beachten Sie dabei, dass das Adapterkabel der Kamera nicht verlängert werden kann, was die Flexibilität des Kamerastandorts einschränkt. Die Reichweite des Bewegungssensors (PIR) ist relativ groß; hier sind Sie deutlich flexibler, was die Positionierung des Sensors in Ihrer Wohnung angeht.

Schließen Sie jetzt die Batterie an. Der Pi Zero startet und lädt das Script. Lassen Sie den Pi den Tag über laufen. Anschließend schließen Sie ihn an einen Monitor an, stoppen das Script und laden mit **startx** die grafische Benutzeroberfläche. Jetzt können Sie auf die Bilder zugreifen.

spy.py

Sprache
>PYTHON 3

DOWNLOAD:
magpi.cc/PiSpy

```
#!/usr/bin/env python

from gpiozero import MotionSensor
from picamera import PiCamera
from datetime import datetime
from time import sleep

sensor = MotionSensor(14)
camera = PiCamera()

while True:
    sensor.wait_for_motion()
    filename = datetime.now().strftime("%H.%M.%S_%Y-%m-%d.jpg")
    camera.capture(filename)
    sleep(5)
```

SEAN MCMANUS

Sean McManus ist der Autor des Buchs *Raspberry Pi For Dummies* (zusammen mit Mike Cook) und *Scratch Programming in Easy Steps*.
sean.co.uk

Scratch liefert Ihnen eine Auswahl an Spielfiguren frei Haus. Dazu gehört auch dieser kleine putzige Igel, der in unserem Scratch-Workshop am Schluss auf dem Trampolin herumspringt

Sie müssen flott reagieren, um den Igel rechtzeitig mit dem Trampolin zu sichern. Sie schieben es mit den Pfeiltasten hin und her. So verhindern Sie, dass der kleine Igel auf dem Boden aufschlägt

SPIELEN MIT SCRATCH

Programmieren lernen und Spaß dabei haben: Dieses Tutorial zeigt Ihnen, wie Sie ein kleines Reaktionsspiel zum Laufen bringen

Man kann sich stundenlang mit der Hardware des Raspberry Pi beschäftigen, allein die GPIO-Pins sind abendfüllend. Doch selbst hoch motivierte Raspberry-Fans brauchen eine Pause. Wie wäre es da mit einem selbst programmierten Spiel – ganz simpel in Scratch? Wir zeigen Ihnen, wie Sie in dieser visuellen Programmiersprache Charaktere, Farben und Hintergründe austauschen, per Drag&Drop Funktionsblöcke zusammenfügen und Ihr Spiel mit Grafiken, Animationen oder Klängen aufpeppen.

>Schritt-01

Projekt vorbereiten

Starten Sie zuerst Scratch (**Entwicklung | Scratch**). Klicken Sie rechts unten mit der rechten Maustaste auf die Katze. Wählen Sie im Kontextmenü **Löschen**. In der Leiste „Neues Objekt“ wählen Sie das Symbol mit dem Ordner/Stern. Fügen Sie aus dem Ordner **Things** das Trampolin hinzu. Aus dem Unterordner **Fantasy** laden Sie das Objekt „fantasy11“. Suchen Sie sich aus dem Verzeichnis **Backgrounds** den passenden Hintergrund für Ihr Spiel heraus. Unser Beispiel stammt aus dem Unterordner **Outdoors** und heißt „atom-playground“. Im Fenster „Neues Objekt“ sollten jetzt drei Grafikvorlagen zu sehen sein.

.01

>Schritt-02

Script schreiben

.02

Wählen Sie das **Objekt1** aus (Trampolin). Ziehen Sie links aus dem Fenster den Befehlsblock **Wenn angeklickt** in das Editor-Fenster. Klicken Sie auf das Trampolin, dann auf den Tabulator **Scripte**. Weiter oben auf dieser Seite sehen Sie das **Listing .01**. Orientieren Sie sich an diesem Listing und fügen Sie die Befehlsblöcke nach dem gleichen Muster aneinander. Die Werte in den weißen Feldern füllen Sie genauso aus, wie es in den Bildschirmfotos zu sehen ist.

.03

>Schritt-03

Igel-Position festlegen

Klicken Sie auf den Igel. Fügen Sie das Script aus dem **Listing .02** hinzu. Damit wird die Figur links oben auf die Bühne (Abspielfenster) gesetzt, sodass der Spieler eine Chance hat, die Figur rechtzeitig zu entdecken.

.04

>Schritt-04

Einen Loop hinzufügen

Als Nächstes werden wir das Script erweitern, indem wir weiter unten zusätzliche Befehlsblöcke hinzufügen. Werfen Sie einen Blick auf **Listing 03**. Fügen Sie den Block **`wiederhole bis`** in das Script ein. Dann setzen Sie den Operator **`<`** (grüner Block aus der Befehlsliste) in die diamantförmige Leerstelle hinter „wiederhole bis“ ein. Rechts in die Box tragen Sie den Wert **`-120`** ein.

Objekte laden Mit einem rechten Mausklick auf dieses Symbol greifen Sie auf den Inhalt der Objekt-Bibliothek zu

Die verbleibende Leerstelle füllen Sie dann mit dem Blockbefehl **`y-Position`** (blaue Farbe, ganz unten im Abschnitt „Bewegung“). Alles innerhalb der Schleife **`wiederhole bis`** wird solange wiederholt, bis die Position der Spielfigur einen ganz bestimmten y-Wert erreicht hat. In unserem Spiel bedeutet dies, dass die Spielfigur auf dem Boden aufschlägt, falls sie das Trampoline verpasst. Das gilt es beim Spielen zu verhindern.

Operatoren Wenn Sie zum Beispiel Zufallszahlen oder logische Operatoren benötigen, wählen Sie einen der grünen Befehlsblöcke

>Schritt-05

Spielfigur animieren

Damit sich der Igel rührt, fügen Sie zwei Befehle (Bewegung, blaue Farbe) hinzu, siehe **Listing 4**. Diese setzen Sie in den Block **`wiederhole bis`** ein. Klicken Sie auf die grüne Flagge, um den ersten Testlauf zu starten. Der Igel sollte jetzt oben links im Bildschirm auftauchen, herunterfallen und dann stoppen.

.05

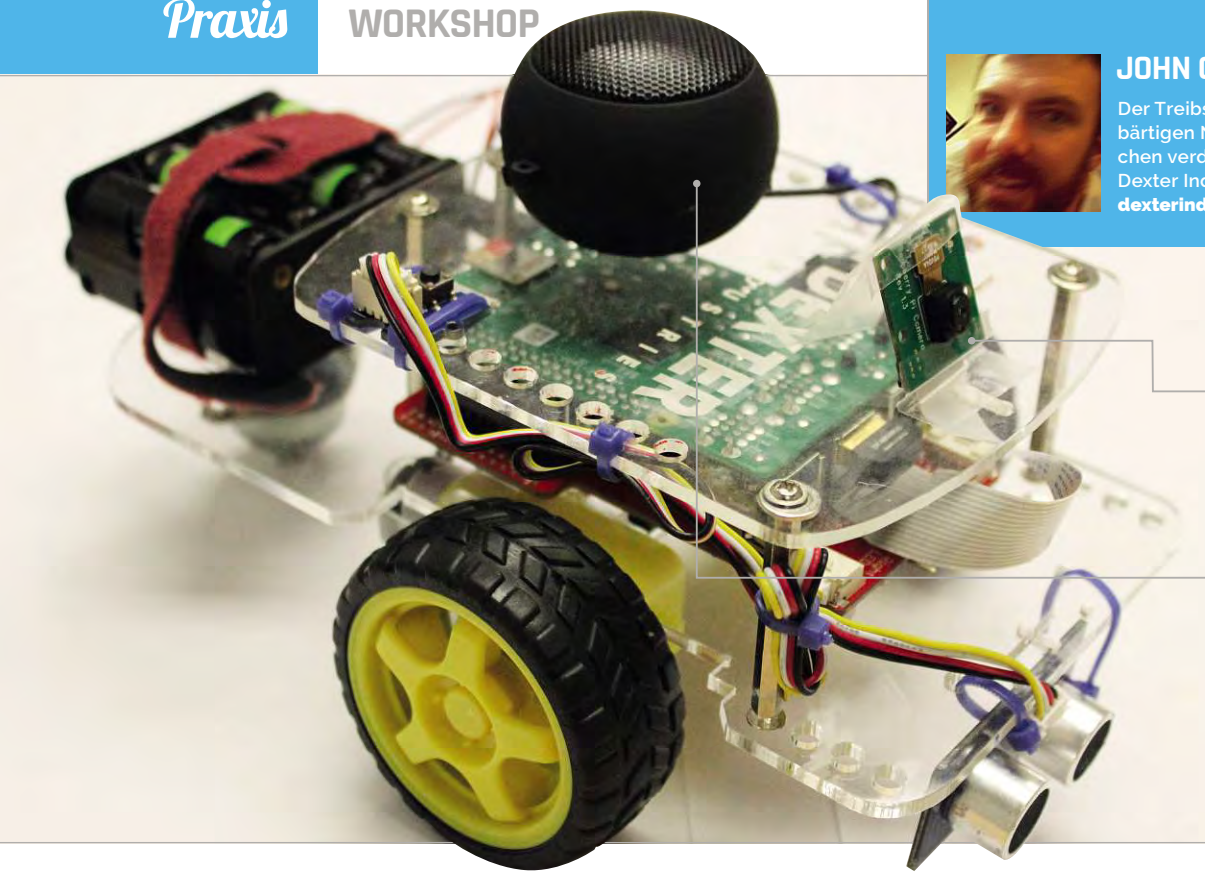
>Schritt-06

Trampolin sprungbereit machen

Nun sorgen wir dafür, dass die Spielfigur vom Trampolin abprallt und in die Höhe katapultiert wird, wenn sie das Trampolin berührt. Ziehen Sie einen **`falls-`**Block in Ihr Script. Positionieren Sie ihn unterhalb von **`pralle vom Rand ab`** und innerhalb von **`wiederhole bis`** (siehe dazu **Listing 05**). Dann ziehen Sie den Block **`wird berührt`** in die Leerstelle des **`falls-`**Blocks. Nun benötigen Sie noch zwei Blöcke aus der Befehlsliste „Bewegung“ sowie „Operatoren“. Es handelt sich zum einen um den Block **`zeige Richtung`** (Bewegung, blau) und zum anderen um **`Zufallszahl von bis`** (Operatoren, grün). Wir verwenden einen Wertebereich zwischen **`-45`** und **`45`**, siehe **Listing 05**. Wenn die Spielfigur nun das Trampolin berührt, prallt sie in einem Winkel von 45 Grad ab. Zum Schluss können Sie das Script um einen Sprach-Block ergänzen: **`Sage Hallo für 2 Sekunden`**. Ändern Sie „Hallo“ zum Beispiel in „Ups, Spiel verloren!“

SCRATCH: BUCHTIPP

Perfekt für den Einstieg und optisch sehr ansprechend gestaltet: „Spiele programmieren super-easy“. ISBN: 3831030952, 16,95 Euro



JOHN COLE

Der Treibstoff für Gehirn und Herz des bärtigen Nerds ist Kaffee. Seine Brötchen verdient er als Ingenieur bei Dexter Industries.
dexterindustries.com

EMPATHYBOT: DER PI-ROBOTER, DER GEFÜHLE ERKENNT

Bei diesem Projekt basteln wir einen Raspberry-Pi-Roboter, der mithilfe des Kameramoduls Emotionen auswerten kann. Er teilt uns mit, ob eine Person glücklich, traurig, verärgert oder überrascht ist

Sie brauchen

- GoPiGo Kit magpi.cc/2hA8f6i
- WiFi-Dongle
- Lautsprecher magpi.cc/2hAgCi7
- Ultraschallsensor magpi.cc/2hAazds
- Tastsensor magpi.cc/2hAhqE1
- Raspberry Pi Kameramodul bit.ly/2kksMuQ

Würde Ihnen ein empathischer Roboter gefallen? In diesem Artikel zeigen wir Ihnen, wie Sie einen solchen Roboter entwickeln, der anhand von Gesichtern Emotionen interpretieren kann. Dafür nutzen wir einen Raspberry Pi, das Pi-Kameramodul, ein GoPiGo-Roboterkit und einen Lautsprecher, der Antworten ausgibt. Unser Roboter kann sich zu einer Person bewegen, ein Foto machen und im Anschluss das Gesicht mit Google Cloud Vision analysieren. Danach äußert sich der Roboter zum Gemütszustand der analysierten Person.

>SCHRITT 01

GoPiGo zusammenbasteln

Dieser Workshop basiert auf dem GoPiGo-Roboterkit (ca. 100 Euro, ohne Raspberry Pi), lässt sich aber auch auf andere Bausätze übertragen. Auf der Webseite von Dexter Industries finden Sie Videos und andere Anleitungen dazu, wie Sie das Roboterkit zusammenbasteln. Bringen Sie den Raspberry Pi an und versorgen Sie den GoPiGo mit acht AA-Batterien. Während der stationären Programmierung ist es allerdings besser, den Pi per Netzteil mit Strom zu versorgen.

>SCHRITT 02

Kamera anbringen

Befestigen Sie das Kameramodul am GoPiGo. In diesem Beitrag verwenden wir den oberen Steckplatz, wie es im Bild oben zu sehen ist.

>SCHRITT 03

Lautsprecher befestigen

Den Lautsprecher können Sie mit ein paar Kabelbindern oben am GoPiGo anbringen. Verbinden Sie ihn danach mit einem entsprechendem Audio-Kabel mit dem Kopfhörerausgang des Pis und laden Sie ihn via USB.

>SCHRITT 04

Sensoren befestigen

Wir benutzen den Tastsensor, um den Roboter leicht starten zu können, sobald der Code läuft. Der Abstandssensor (Ultraschall) verrät, wie weit ein Mensch entfernt ist. Befestigen Sie den Abstandssensor mit Kabelbindern vorne am GoPiGo. Den Taster bringen Sie am besten oben auf dem Gerät an, damit er einfach erreichbar ist.

>SCHRITT 05

Kostenloses Konto bei Google Cloud Vision

Sie können Ihr Gmail- oder Google-Konto für diesen Schritt benutzen. Derzeit bietet Google einen kostenlosen Testzeitraum von 60 Tagen an: **magpi.cc/2hAkotI**.

>SCHRITT 06

Neues Projekt erstellen

Eine Schritt-für-Schritt-Anleitung, wie Sie bei Google Cloud Vision ein neues Projekt erstellen, finden Sie hier: **magpi.cc/2hAhHqc**.

Erstellen Sie ein neues Projekt mit dem Namen „vision1“ und aktivieren die Cloud Vision API dafür.

>SCHRITT 07

JSON-Anmeldedaten herunterladen

Wechseln Sie wieder zur Google Cloud Console. Dort wählen Sie den Bereich **Use Google APIs**. Klicken Sie auf **Enable and manage APIs** und danach auf **Credentials** und **Create Credentials**. Den Button „Credentials“ finden Sie links auf der Seite, danben ist ein Schlüssel abgebildet. Wählen Sie **Create a Service Account Key**. Unter **Service Account** finden Sie **New Service Account**. Wir nennen das Konto **vision**. Nun kreieren Sie noch eine Rolle, die vollständigen Zugriff bekommt. Wählen Sie **Project** und **Owner**, damit der Pi auf alle Ressourcen zugreifen kann. Ein Pop-up-Fenster öffnet sich und teilt mit, dass Sie

Oben Das Projekt haben wir auch mit Kleinkindern getestet. Es funktioniert wunderbar.

Oben Das Projekt funktioniert nur mit bartlosen Menschen. Bärte verwirren die Software ziemlich ...

Sprache

>PYTHON

DOWNLOAD:
magpi.cc/2hAgqiT

einen neuen Schlüssel erzeugt haben. Der Download startet automatisch. Verlieren Sie diese Datei nicht! Mit einem FTP-Program wie FileZilla oder via Samba (Anleitung: **youtu.be/CEYwYqkAPfA**) kopieren Sie die JSON-Datei auf den Raspberry Pi in das Verzeichnis **/home/pi**.

>SCHRITT 08

Raspberry Pi konfigurieren

Wir müssen für die Konfiguration des Pis einige Befehle ausführen. Das ist nur ein einziges Mal erforderlich.Führen Sie Folgendes auf der Kommandozeile aus:

```
sudo pip install --upgrade pip
sudo apt install libjpeg8-dev
sudo pip install --upgrade
google-apipython-client
sudo pip install --upgrade Pillow
sudo apt install python-picamera
```

Dann stellen Sie Python die JSON-Datei zur Verfügung, die Sie zuvor heruntergeladen haben:

```
export GOOGLE_APPLICATION_
CREDENTIALS=Dateiname.json
```

Ersetzen Sie „Dateiname“ mit dem Namen Ihrer Datei.

>SCHRITT 09

Den Code ausführen

Es ist egal, ob Sie den Code abgetippt oder von unserer Heft-DVD kopiert haben. Sie lassen ihn so laufen:

```
sudo python empathybot.py
```

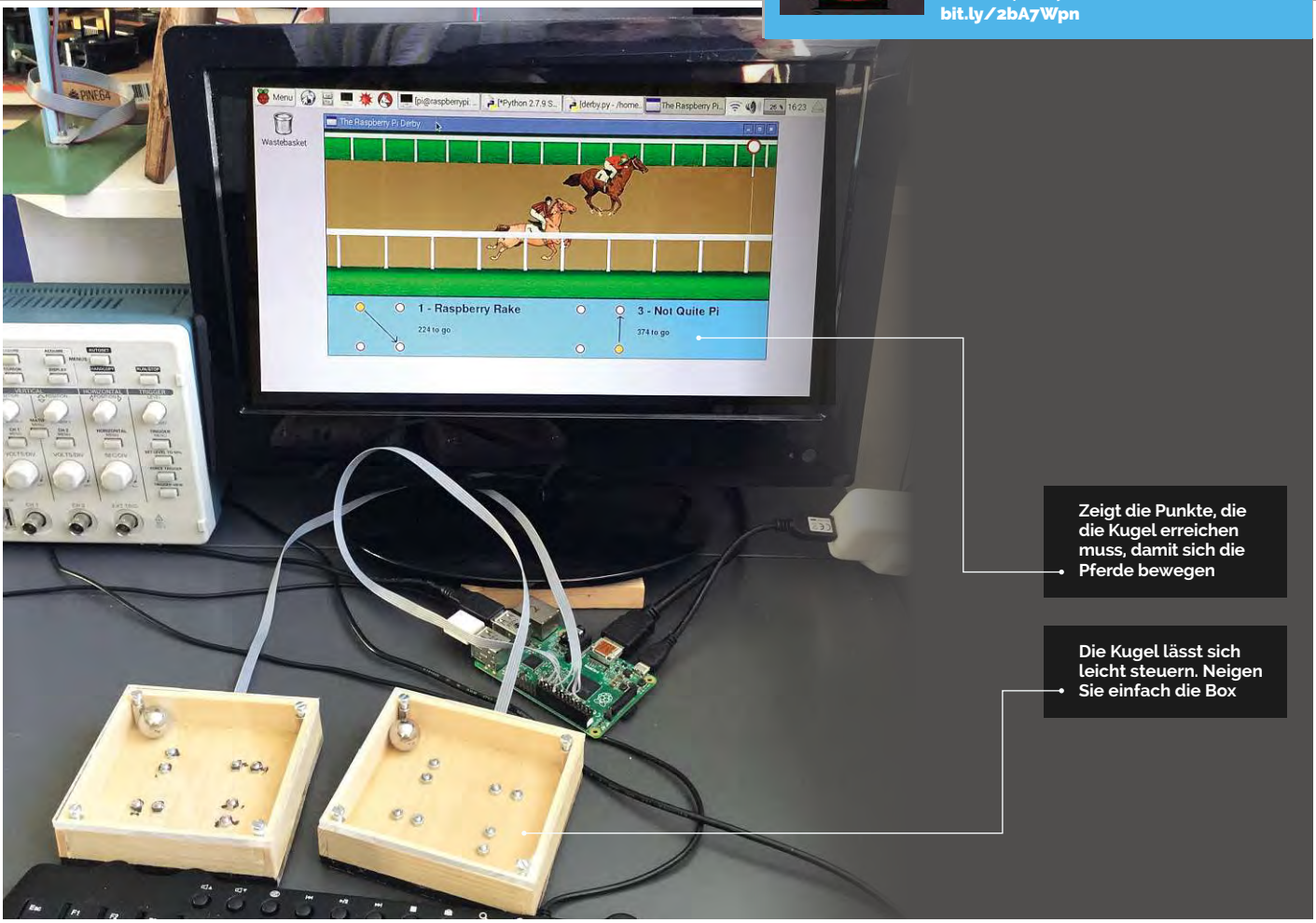
Platzieren Sie den Roboter in Richtung einer Person und drücken die Sie den Tastsensor auf dem GoPiGo. Schon legt der Roboter los!

MIKES PI-PROJEKT

MIKE COOK

gehört zu den altgedienten Autoren. Mike hat schon viele Workshops entwickelt und ist Mitautor von *Raspberry Pi für Dummies* und *Spannende Projekte mit dem Raspberry Pi*.

bit.ly/2bA7Wpn



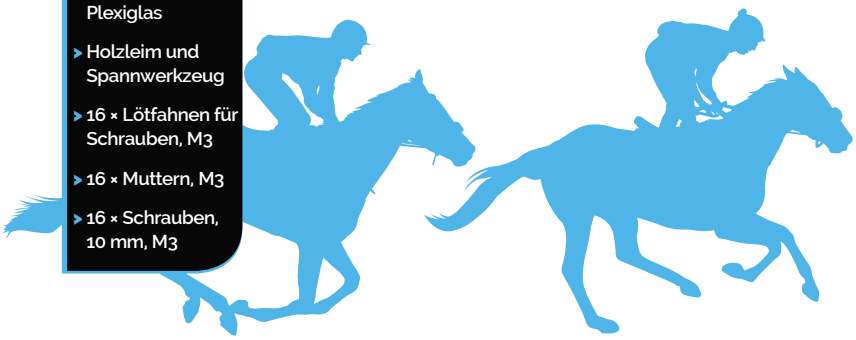
Sie brauchen

- 2 × 16 mm Metallkugeln
- 8 × 15 mm, M3 Abstandsbolzen
- Diverse Kabel sowie Lötzinn und Lötkolben
- Sperrholz und Acryl- bzw. Plexiglas
- Holzleim und Spannwerkzeug
- 16 × Lötfahnen für Schrauben, M3
- 16 × Muttern, M3
- 16 × Schrauben, 10 mm, M3

RENNFIEBER

EIN KIRMES-KLASSIKER LÄUFT AUF DEM RASPI

Wer überquert in diesem Zwei-Personen-Spiel als Erster die Ziellinie?



Kirmesbesucher lieben es: das spannende Pferderennen, bei dem eine kleine Kugel über Sieg oder Niederlage entscheidet. Wer es schon einmal auf dem Jahrmarkt gespielt hat, kennt das Prinzip. Jeder Spieler bekommt einen Ball, den man über eine schräge Bahn in das passende Loch schubst. Je nachdem, welches man erwischt, sprintet das Pferd in gewaltigen Sätzen nach vorne oder verharrt auf der Stelle. Genau dieses Spiel wollen wir auf dem Raspberry Pi in diesem Workshop simulieren.

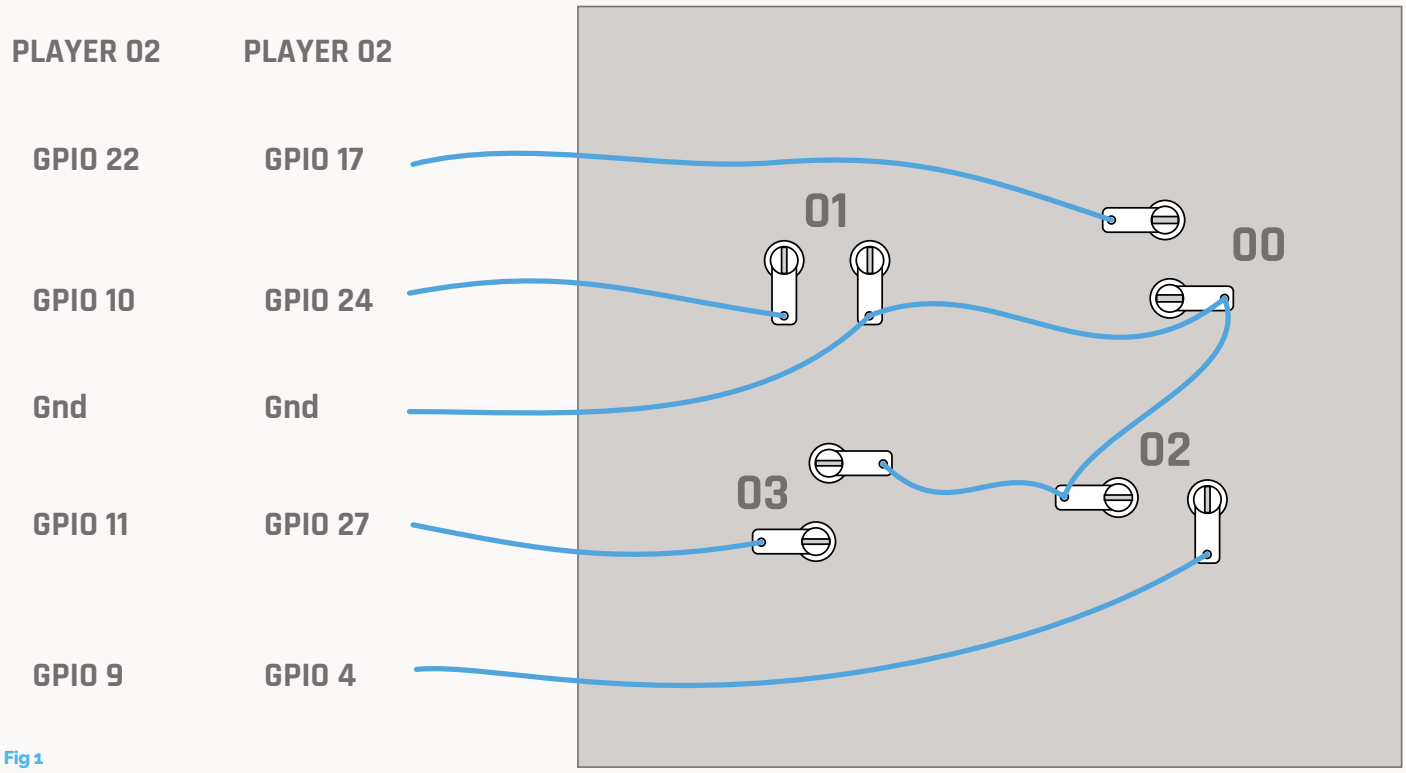


Fig 1

Unser Spielaufbau

Es versteht sich von selbst, dass man einige Änderungen vornehmen muss, damit das ursprüngliche Kirmesspiel auf dem RasPi läuft. Die ersten Maßnahmen: Alle Konstruktionsteile schrumpfen auf eine handliche Größe; die Zahl der Spieler ist auf zwei Personen beschränkt. Die mechanisch angetriebenen Pferde ersetzen wir durch virtuelle Galopper, ebenso wie die weitläufige Rennbahn – beides wird auf dem Monitor dargestellt. Als Controller verwenden wir zwei Holzkisten, in denen kleine Stahlkugeln geschickt zu kleinen Metallpfosten manövriert werden. Das Prinzip ähnelt den klassischen Kugellabyrinthen, bei denen man kleine Kügelchen in Löcher versenken muss. Unser Augenmerk haben wir bei der Rennsimulation auf die Geschicklichkeit gelegt, der Faktor Zeit ist nebensächlich.

Die Controller-Boxen

In der Controller-Box befinden sich vier metallische Kontaktpaare. Sobald die Stahlkugel beide Kontakte gleichzeitig berührt, wird der Stromkreis geschlossen. Die beiden Metallpfosten dienen also als separate Schalter. Die Pfosten sind mit GPIO-Pins verbunden, das erlaubt die Auswertung per Software. Die Verkabelung sehen Sie in der obigen Abbildung (**Fig 1**). Wenn Sie sich die Anordnung der Kontakte ansehen, fällt Ihnen sicherlich auf, dass die Pfosten unterschiedlich ausgerichtet sind. Das hat einen einfachen Grund: Dadurch ist es viel schwieriger, von einem Pfostenpaar zum anderen zu gelangen – was mehr Geschicklichkeit erfordert. Die Konstruktionsdetails entnehmen Sie der Schritt-für-Schritt-Anleitung.

CONTROLLER BAUEN

>SCHRITT-01

Material zusammenstellen

Für die Boxen benötigen Sie jeweils zwei Platten mit 6 mm dickem Sperrholz in den Abmessungen 100 mm x 90 mm. Die Abdeckung fertigen Sie passend aus Acrylglas bzw. Kunststoff. Für die Seitenteile schneiden Sie acht Holzstücke zurecht (eventuell aus einer anderen Holzart).

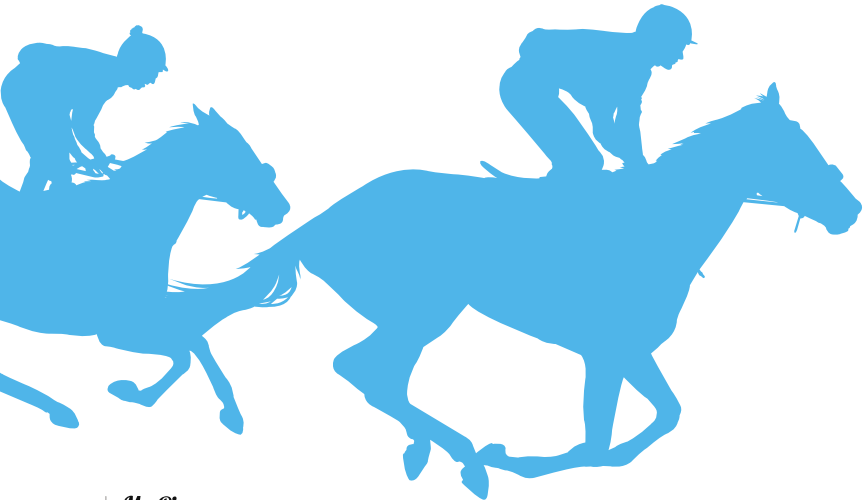
>SCHRITT-02

Der Boden der Holzbox

Für geübte Handwerker ist der Zusammenbau der kleinen Holzboxen (Controller) kein Problem. Wir empfehlen Ihnen, ein Spezialwerkzeug wie zum Beispiel einen Rahmenspanner zu verwenden (am besten mit Spannbacken, siehe Foto). Dann können Sie sicher sein, dass die Seitenteile fest zusammengedrückt werden und der Holzleim ungestört aushärtet.

Die grafische Gestaltung

Der sicherlich aufwendigste Teil ist die optische Aufmachung des Spiels. Stimmungsvoll soll es sein und an die typische Atmosphäre eines Galopprennens erinnern. Dazu haben wir uns aus alten Zeitschriften schöne Abbildungen mit Rennsportpferden besorgt, diese ausgeschnitten und per Scanner digitalisiert. So können wir sie als Sprites (Objekte) in das Spiel importieren und später über den Bildschirm bewegen.



Der Hintergrund dieser Sprites ist freigestellt. Dazu benötigt man ein Grafik- bzw. Bildbearbeitungsprogramm, etwa GIMP. Anschließend haben wir die beiden Grafikobjekte auf eine Größe von 196 x 136 Pixel verkleinert. Die Hintergrundgrafik, also die Rennbahn, wurde ebenfalls per Software gestaltet. Bei der Streckenabsperrung haben wir darauf geachtet, dass die Zwischenräume des Zauns transparent sind. Dann wurde noch eine Grafik erstellt, die den Pfad der Kugel und die Start- und Zielpunkte zeigt. Um das Ganze abzurunden, haben wir uns im Internet einige passende Soundschnipsel besorgt, um die Rennatmosphäre möglichst naturgetreu zu simulieren.

Die Spielregeln

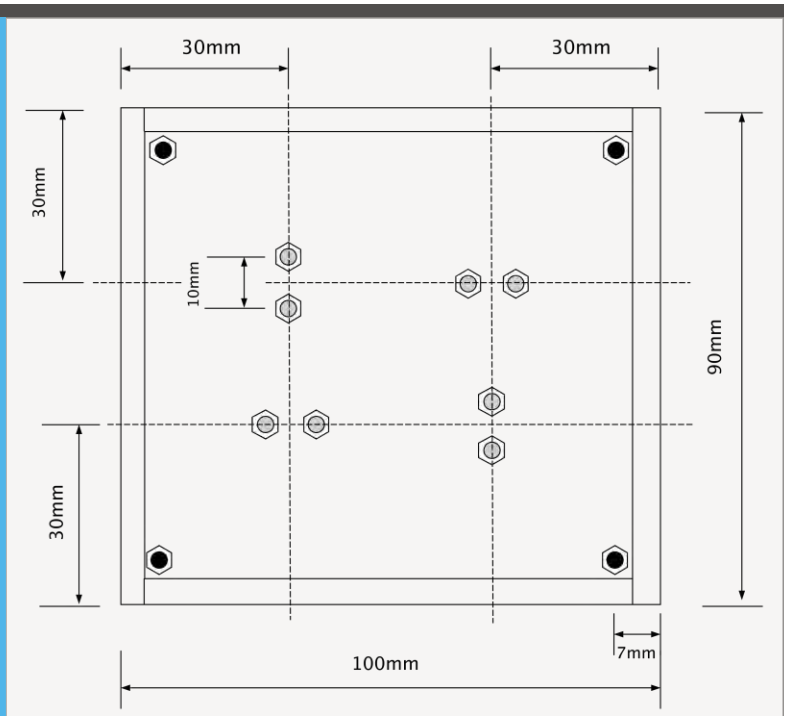
Die Bildschirmgrafik oben auf der folgenden Seite („Start-Ziel“) zeigt die beiden Schalter, die der Ball berühren muss, um das jeweilige Pferd in Bewegung zu setzen. Der orangefarbene Kreis markiert die Startposition. Nur wenn Sie die Kugel auf „Start“ setzen und sie dann zum Ziel bewegen, galoppiert Ihr Pferd in Richtung Ziellinie. Sobald die Kugel auf „Start“ liegt, erklingt ein fröhliches Wiehern, wenn die Kugel den nächsten Schalter berührt, hören Sie den Pferde-

Start-Ziel Bewegen Sie den Ball in die Richtung, die jeweils auf der Grafik zu sehen ist. Den Startpunkt markiert der orangefarbene Kreis

galopp. Gleichzeitig erscheint auf dem Bildschirm eine Pferdeanimation. Sobald das Rennen startet, galoppiert das Pferd nicht nur in Richtung der Ziellinie, sondern bewegt sich gleichzeitig auf und ab, um den Galopp des Originalspiels möglichst naturgetreu zu simulieren. Sobald eines der beiden Pferde die Ziellinie als Sieger überquert, wird der Vorsprung berechnet, mit dem es gewonnen hat. Anschließend folgt die Startaufstellung für das nächste Rennen. Alternativ drücken Sie die Eingabe-Taste. Dann startet das Rennen sofort neu – auch mitten im Lauf.

Die Software fürs Spiel

Den Code für das Spiel, sprich **derby.py**, finden Sie auf den folgenden Seiten als Listing. Sie können ihn von unserer **Heft-DVD** oder von der GitHub-Seite **magpi.cc/1NqJjmV** („Horse_Derby“) herunterladen. Im Unterverzeichnis **Horse_Derby/software/images** finden Sie alle für das Rennspiel nötigen Grafiken. Die Audioschnipsel sind im Ordner **sounds** abgelegt. Wie viele unserer Projekte basiert auch dieses Spiel auf dem Pygame-Framework, was es erleichtert, den Code zu lesen. Beispiel: Die Funktion **`showPicture`** zeichnet zuerst den Pfad und blendet dann die beiden Pferde ein, gefolgt von den Spuren, auf denen sich die Pferde bewegen. Das Programm berechnet übrigens nach dem Überqueren der Ziellinie nicht nur den nummerischen Abstand des Siegers gegenüber dem Verlierer. Es verwendet auch die Fachbegriffe aus der Pferderennsportszene, um die Leistung des Siegers zu charakterisieren. Dazu einige Beispiele: „a distance“, „a neck“ oder „a nose“. Wenn Sie die englischsprachige Bezeichnung stört, müssen Sie den Code ab Zeile 172 ändern. Das Gleiche gilt, wenn Sie für das Spiel die Namen Ihrer Lieblingspferde verwenden möchten, siehe ab Zeile 149 folgende. Noch ein Wort zur Variable **`movePhase`**: Während das Spiel läuft, überprüft das Programm, an welcher Position sich die Kugel befindet. Immer, wenn sich das Pferd bewegt, kontrolliert die Software, ob der Galopper das Ziel bereits überschritten hat. Auch für den unwahrscheinlichen Fall, dass beide Pferde die Ziellinie gleichzeitig überqueren, haben wir im Code vorgesorgt: Dann gibt das Programm den Hinweis „dead heat“, also „Totes Rennen“ aus (siehe Zeile 164).



>SCHRITT-03

Bohrungen anbringen

Das obige Diagramm zeigt den Blick von oben in die Holzbox. Sie finden es auch unter den Projektdateien auf der Heft-DVD (HowTo3.pdf). Es gibt die Punkte an, die für die Bohrungen vorgesehen sind. Tipp: Drucken Sie es sich aus und verwenden Sie die Zeichnung als Bohrschablone. Anschließend verkabeln Sie alles und löten die Lötfahnen an die Schrauben. Sie dienen später als Anschlüsse für die GPIOs des RasPi.

>SCHRITT-04

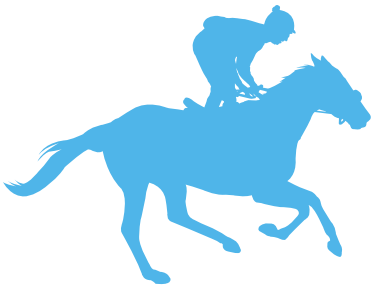
Der letzte Schliff

Achten Sie darauf, dass alle Verbindungen sauber sind. Reinigen Sie die Schrauben und die Stahlkugeln. Schon etwas Fingerfett darauf beeinträchtigt unter Umständen den elektrischen Kontakt. Sie können die Schrauben auch sicherheitshalber mit leitfähiger Farbe bestreichen.

derby.py

```
# Pi Derby - Horse race game
# By Mike Cook - November 2016

import pygame, time, os, random
import wiringpi2 as io

pygame.init()  # initialise graphics interface
pygame.mixer.quit()
pygame.mixer.init(frequency=22050, size=-16,
channels=2, buffer=512)

os.environ['SDL_VIDEO_WINDOW_POS'] = 'center'
pygame.display.set_caption("The Raspberry Pi Derby")
pygame.event.set_allowed(None)
pygame.event.set_allowed([pygame.KEYDOWN,pygame.QUIT])
screen = pygame.display.set_mode([942,466],0,32)
textHeight = 36 ; textHeight2 = 24
font = pygame.font.Font(None, textHeight)
font2 = pygame.font.Font(None, textHeight2)

random.seed() ; winningPost = 704
ballPins = [ [17,24,4,27],[22,10,9,11 ] ]
targetGx = [ 18,102,18,102 ]
targetGy = [ 18,18,102,102 ]
gallopInc = [1,1] ; gallop= [0,0]
dInc = [0,0] # distance increment
puzzle = [5,5] ; restart = True
moveTarget = [1,1] ; movePhase = [0,0]

def main():
  global restart,dInc,gallop,puzzle,moveTarget,movePhase
  initGraphics()
  initGPIO()
  #print "The Pi Derby"
  distance = [-120, -120]
  showPicture(distance,gallop)
  while True:
    checkForEvent()
    if restart :
      gameSound[3].play()
      windBack(distance)
      pygame.mixer.fadeout(1000)
      time.sleep(3.5)
      distance[0] = -120 ; distance[1] = -120
      gallop[0] = 0 ; gallop[1] = 0
      gallopInc[0] = 1 ; gallopInc[1] = 1
      dInc[0] = 0 ; dInc[1] = 0
      puzzle = [5,5]
      moveTarget = [1,1] ; movePhase = [0,0]
      gameSound[0].play()
      showPicture(distance,gallop)
      time.sleep(5)
      restart = False # show Puzzle
      showPicture(distance,gallop)

    moveDetect(distance)
    if dInc[0] or dInc[1] :
      for n in range(0,2):
        if dInc[n] :
          distance[n] += 3
          if dInc[n] :
            dInc[n] -=1
            gallop[n] = gallopAdv(gallop,n)
      showPicture(distance,gallop)
      if checkForFinish(distance):
        restart = True
        time.sleep(4.0)

def moveDetect(dis):
  global dInc, puzzle, moveTarget
  move = -1
  for n in range(0,2):
    move = checkInput(n) # look at ball switch
    if move != -1 :
    # check if it is a valid move
    if move ==  moveTarget[n] :
      if movePhase[n] == 1 : # move complete
        dInc[n] = 50 # move this * 3
        gameSound[1].play()
        last = moveTarget[n]
        puzzle[n] = random.randint(0,11)
        while last == moveState[puzzle[n]][0] :
          puzzle[n] = random.randint(0,11)
        moveTarget[n] = moveState[puzzle[n]][0]
        movePhase[n] = 0
      else :
        moveTarget[n] = moveState[puzzle[n]][1]
        movePhase[n] = 1
        showPicture(dis,gallop) # update move graphic
        if n ==0 :
          gameSound[4].play()
        else :
          gameSound[5].play()

def gallopAdv(gallop,n):
  global gallopInc
  gallop[n] += gallopInc[n]
  if gallop[n] > 6 or gallop[n] <0:
    gallopInc[n] = -gallopInc[n]
  return gallop[n]

def checkForFinish(d):
  if d[0] >=winningPost or d[1] >=winningPost:
    gameSound[2].play()
    finish(d[0] - d[1])
    time.sleep(3.0)
    return True
  else :
    return False

def windBack(d):
  wind = 4
  while d[0] >= -120 or d[1] >= -120:
    showPicture(d,gallop)
    checkForEvent()
    for n in range(0,2):
      if d[n] >=-120:
        d[n] -= wind

def checkInput(player):
  ball = -1
  for pin in range(0,4):
    if io.digitalRead(ballPins[player][pin]) == 0 :
      ball = pin
  return ball
```

```
125.
126. def initGPIO():
127.   try :
128.     io.wiringPiSetupGpio()
129.   except :
130.     print"start IDLE with 'gksudo idle' from command line"
131.     os._exit(1)
132.   for player in range(0,2):
133.     for pin in range (0,4):
134.       io.pinMode(ballPins[player][pin],0)
135.       # input enable pull up
136.       io.pullUpDnControl(ballPins[player][pin],2)
137.
138. def showPicture(run,gal):
139.   screen.blit(background,[0,0])
140.   screen.blit(horse1,[run[0],30+gal[0] ] )
141.   screen.blit(horse3,[run[1],120+gal[1] ] )
142.   screen.blit(foreground,[0,200])
143.   pygame.draw.rect(screen,(102,204,255), (0,340,942,136), 0)
144.   if not restart :
145.     screen.blit(move[puzzle[0]],[50,345])
146.     pygame.draw.circle(screen, (255,154,51),(
    targetGx[moveTarget[0]]+50,targetGy[moveTarget[0]]+345),8,0)
147.     screen.blit(move[puzzle[1]],[471+50,345])
148.     pygame.draw.circle(screen, (255,154,51),(
    targetGx[moveTarget[1]]+50+471,targetGy[moveTarget[1]]+345),8,0)
149.   drawWords("1 - Raspberry Rake",120+70,350,0)
150.   drawWords("3 - Not Quite Pi",120+471+70,350,0)
151.   drawWords(str(winningPost-run[0])+" to go",120+70,400,1)
152.   drawWords(str(winningPost-run[1])+" to go",120+70+471,400,1)
153.   pygame.display.update()
154.
155. def finish(distance):
156.   if distance != 0:
157.     caption = "Wins by " + margin(abs(distance))
158.     if distance < 0 :
159.       x= 120+70+471
160.     else:
161.       x= 120+70
162.     drawWords(caption,x,400,1)
163.   else :
164.     drawWords("Dead heat",120+70,400,1)
165.     drawWords("Dead heat",120+70+471,400,1)
166.   pygame.display.update()
167.
168.
169. def margin(dist):
170.   separation = dist / 120.0
171.   if separation >1.8:
172.     return "a distance"
173.   elif separation >0.9:
174.     return "a length"
175.   elif separation >0.7:
176.     return "three quarters of a length"
177.   elif separation >0.4:
178.     return "half a length"
179.   elif separation >0.25:
180.     return "a neck"
181.   elif separation >0.2:
182.     return "a short neck"
183.   elif separation >0.1:
184.     return "a head"
185.   elif separation >0.05:
186.     return "a short head"
187.   else :
188.     return "a nose"
189.
190. def drawWords(words,x,y,f) :
191.   if f == 0:
192.     th = textHeight
193.   else :
194.     th = textHeight2
195.   textSurface = pygame.Surface(
    (14,th))
196.   textRect = textSurface.get_rect()
197.   textRect.left = x
198.   textRect.top = y
      pygame.draw.rect(screen,(
199. 102,204,255),(x,y,14,th-10), 0)
200.   if f==0 :
        textSurface = font.render(words, True, (0,0,0), (102,204,255))
201.   else :
202.     textSurface = font2.render(
    words, True, (0,0,0), (102,204,255))
203.   screen.blit(textSurface,textRect)
204.
205. def initGraphics():
206.   global background,foreground,horse1,horse3,move,moveState,
    gameSound
207.   soundEffects = ["start","gallop","end","ratchet","horse1",
    "horse2"]
208.   background =  pygame.image.load(
    "images/track.png").convert_alpha()
209.   foreground =  pygame.image.load(
    "images/rail.png").convert_alpha()
210.   horse1 =  pygame.image.load("images/rr.png").convert_alpha()
211.   horse3 =  pygame.image.load("images/np.png").convert_alpha()
212.   move = [ pygame.image.load(
    "images/dir"+str(m)+".png").convert_alpha() for m in range(0,12) ]
213.   moveState = [ [0,3],[3,0],[2,1],[1,2],[0,1],[1,0],[0,2],[2,3],
    [3,1],[1,3],[3,2],[2,0] ]
214.   gameSound = [ pygame.mixer.Sound(
    "sounds/"+soundEffects[sound]+".wav") for sound in range(0,6)]
215.
216. def terminate(): # close down the program
217.    print "Closing down please wait"
218.    pygame.mixer.quit()
219.    pygame.quit() # close pygame
220.    os._exit(1)
221.
222. def checkForEvent(): # see if we need to quit
223.    global restart
224.    event = pygame.event.poll()
225.    if event.type == pygame.QUIT :
226.      terminate()
227.    if event.type == pygame.KEYDOWN :
228.      if event.key == pygame.K_ESCAPE :
229.        terminate()
230.      if event.key == pygame.K_RETURN :
231.        restart = True
232.
233. # Main program logic:
234. if __name__ == '__main__':
235.    main()
```



FRAGEN & ANTWORTEN

In dieser Ausgabe erfahren Sie alles Wichtige über die GPIO

RUND UM DIE GPIO

WOZU DIENT DIE GPIO?

Elektronische Komponenten einbinden

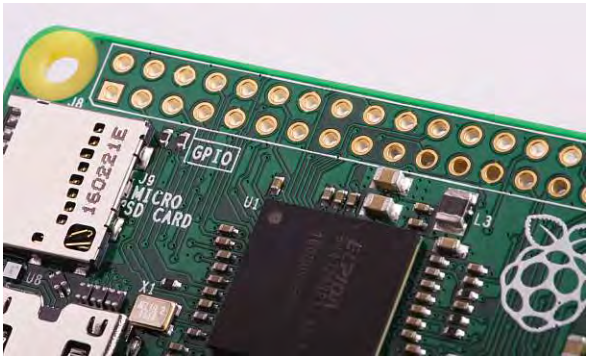
Die GPIO-Pins werden dazu verwendet, den RasPi mit der „realen" Welt zu verbinden. Komponenten wie Sensoren, Lampen oder etwa Signalgeber können auf diese Weise über den Raspberry Pi gesteuert werden.

Add-ons für Raspberry Pi

Außerdem werden die Pins der GPIO (General Purpose Input/Output) eingesetzt, um die Funktionalität des RasPi zu erweitern. Früher wurden oft LC-Displays via GPIO mit dem RasPi verbunden. Heute sind HATs sehr beliebt. Mit diesen Add-ons können Sie Ihren RasPi richtig aufbohren.

Schnittstelle für Geräte

Häufig dient die GPIO auch dazu, externe Geräte anzuschließen. So können Sie etwa Controller einbinden, um Retro-Games zu spielen – oder aber Sie nutzen die GPIO schlicht dazu, Ihren Raspberry Pi mit Strom zu versorgen.



Rechts Der Pi Zero bringt eine GPIO ohne Steckerleiste mit

WOFÜR SIND DIE EINZELNEN PINS DA?

Pins für die Stromversorgung

Einige der 40 Pins dienen der Stromversorgung (5 Volt oder 3,3 Volt), andere können als Masse-Pin eingesetzt werden. Viele weitere Pins können so programmiert werden, dass sie 3,3 Volt liefern (siehe Seite 54).

Frei programmierbare Pins

Die große Mehrheit der Pins kann für unterschiedliche Dinge genutzt werden. Meist steuern sie einfache Komponenten wie LEDs und Signalgeber.

Pins für Spezialaufgaben

Manche der Pins bieten auch Spezialfunktionen: Sie dienen als serielle Schnittstelle (SPI), man kann damit den I²C-Bus des RasPi ansteuern und vieles mehr. Diese Pins werden in anspruchsvolleren Workshops häufig genutzt. Im MagPi informieren wir Sie natürlich, welche Pins Sie genau verwenden müsssen.

WIE VIELE PINS GIBT ES EIGENTLICH?

Die frühen Modelle

Ursprünglich hatte der Raspberry Pi 26 GPIO-Pins, inklusive der Strom-Pins, der Masse-Pins und der frei programmierbaren Pins. Am Markt sind immer noch einge Add-ons für diese Pis erhältlich, sodass man sie noch verwenden kann.

B+ und aufwärts

Aktuelle Pi-Modelle haben 40 Pins – egal ob es sich um einen Pi Zero (allerdings ohne Steckerleiste) oder den Pi 3 handelt. Dabei sind die ersten 26 Pins genauso konfiguriert wie bei den Ursprungsmodellen, sodass ältere Projekte noch funktionieren. Die 14 zusätzlichen Pins sind frei programmierbar oder Masse-Pins.

Compute Module

Für manche Projekte im professionellen Umfeld ist das Compute Modul die beste Wahl. Das I/O-Board hat 200 GPIO-Pins – schließlich muss dieses Modell mehr leisten als nur Lämpchen zum Leuchten zu bringen.

AUS DER RASPBERRY PI FAQ
RASPBERRYPI.ORG/HELP

Welche Performance bietet der Raspberry Pi?
Die Leistung der RasPi-Modelle A, A+, B and B+ ist mit einem 300-MHz-Pentium 2 vergleichbar, aber mit deutlich besserer Grafik – die maximale Anzahl der zu verarbeitenden Pixel bzw. Texturexelemente liegt bei 1 Gpixel/s bzw. 1,5 Gtexel/s, die Rechenleistung liegt bei 24 GFLOPs. Der RasPi 2B entspricht in etwa einem Athlon Thunderbird, der mit 1,1 GHz getaktet ist, aber auch hier schneidet die Grafik besser ab. Modell 3B wiederum ist etwa 50 Prozent schneller als Modell 2B.

Kann man ihn übertakten?
Die Modelle A, A+, B, und B+ sind standardmäßig mit 700 MHz getaktet; in der Regel kann man sie problemlos mit 800 MHz betreiben. Der Pi 2B läuft standardmäßig mit 900 MHz, verkraftet aber auch 1.000 MHz. Das aktuelle Raspbian enthält die Möglichkeit, per `sudo raspi-config` die Overclocking-Optionen zu verändern, ohne den Garantieanspruch zu verlieren. Dennoch sollten Sie beachten, dass sich diese Einstellungen eher für Experimentierfreudige eignen. Es ist nicht gesichert, dass Ihr Board jederzeit stabil läuft, wenn Sie die höchstmöglichen Taktraten einstellen.

Muss der Prozessor gekühlt werden?
Nein, üblicherweise nicht. Der Chip, der im Raspberry Pi verbaut ist, entspricht dem eines Smartphones. Er sollte also nicht so heiß werden, dass er eine spezielle Kühlung benötigt. Je nachdem, was für ein Gehäuse Sie nutzen und wie stark Sie den RasPi übertakten, kann eine Kühlung aber dennoch sinnvoll sein, vor allem, wenn es sich um das Modell 3B dreht.

Welche Hardware-Schnittstellen hat der RasPi?
Je nach Modell hat der RasPi 40 oder 26 GPIO-Pins. Sie alle bieten eine UART-Schnittstelle, einen I²C-Bus, einen SPI-Bus, I²S Audio, 3,3 V, 5V, und Masse. Theoretisch kann die GPIO per I²C- oder SPI-Bus beliebig erweitert werden.

IMPRESSUM

Redaktionsleitung Thorsten Franke-Haverkamp (verantwortlich für den redaktionellen Inhalt)
Chefin vom Dienst Julia Schmidt
Redaktion Russell Barnes, Laura Clay, Patrick Dörfel, Lucy Hattersly, Thorsten Franke-Haverkamp, Phil King, Lorna Lynch, Angelika Reinhard, Julia Schmidt, Rob Zwetsloot
Text-/Schlussredaktion Birgit Lachmann, Angelika Reinhard
Red. Mitarbeit Jürgen Donauer, Dr. Matthias Kampmann, Jörg Reichertz, Matthias Semlinger
Autoren und Entwickler Wesley Archer, Johannes Bergs, Brian Brandaw, Henry Budden, John Cole, Mike Cook, Lauren Egts, Johannes Jägers, James Hobro, Wendell Kapustiak, Matt Long, Sean McManus, Mark Nichols, Jillian Ogle, Clodagh O'Mahony, Matt Richardson, Bruce Shapiro, Masato Sudo
Art Direction Dougal Matthews, Stephanie Schönberger
Grafikleitung Antje Küther
Grafik Veronika Zangl (verantw.), Sam Alder, Lee Allen, Daiva Bumelyte, Andrea Graf, Mike Kay
DVD Karsten Bunz, Patrick Dörfel

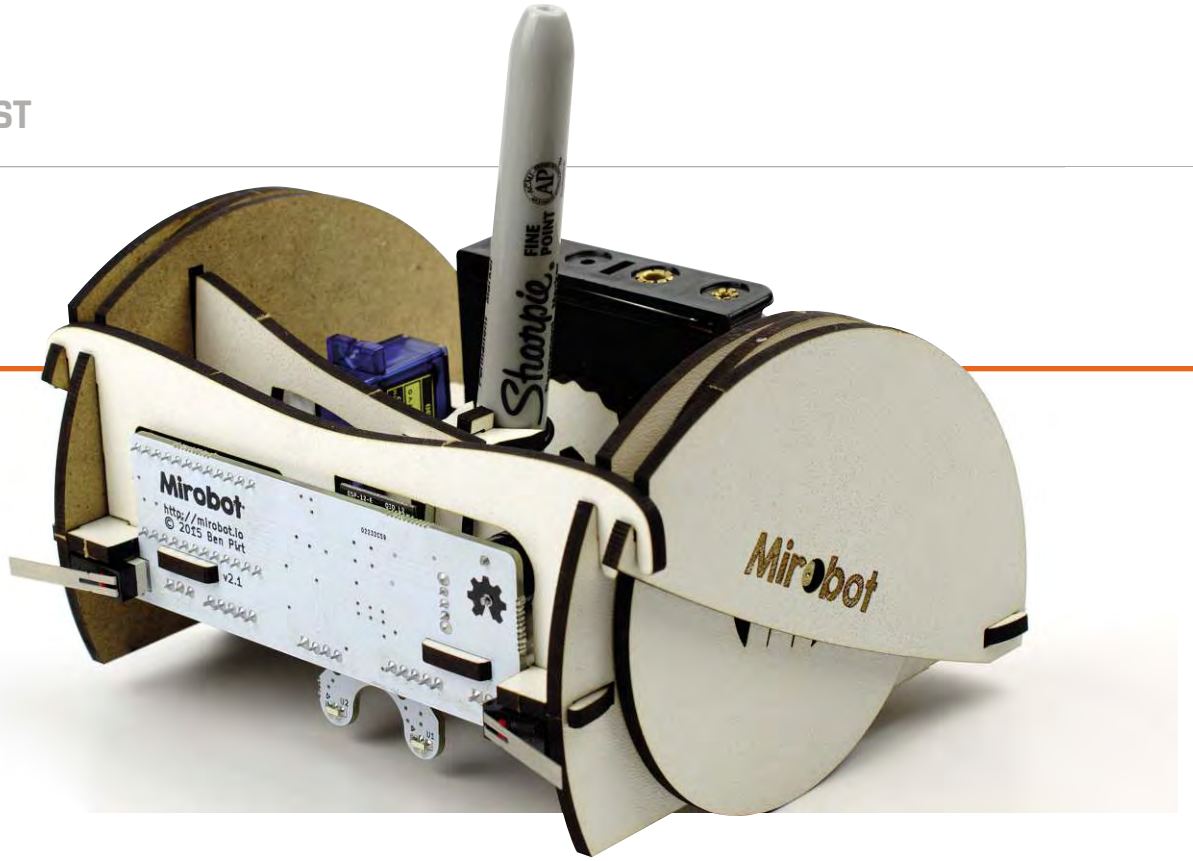
VERLAG UND REDAKTION
Anschrift CHIP Communications GmbH, St.-Martin-Straße 66, 81541 München
Tel. (089) 9250-4590 (Redaktion), -4413 (Fax)
Die Inhaber- und Beteiligungsverhältnisse lauten wie folgt: Alleinige Gesellschafterin ist die Burda Tech Holding GmbH mit Sitz in der St.-Martin-Straße 66, 81541 München
Geschäftsführer Thomas Koelzer (CEO), Markus Scheuermann (COO)
Verleger Prof. Dr. Hubert Burda
Executive Director Florian Schuster
Director Sales Erik Wicha, Tel. (089) 9250-2326, Fax -4542, ewicha@chip.de, chip.de/media
Key Account Manager Katharina Lutz, Tel. -1116, kalutz@chip.de

Verantwortlich für den Anzeigenteil AdTech Factory GmbH & Co. KG, Hauptstraße 127, 77652 Offenburg
Gudrun Nauder, Tel. (089) 9250-2132, gudrun.nauder@adtechfactory.com
Herstellung Andreas Hummel, Frank Schormüller, Medienmanagement, Vogel Business Media GmbH & Co. KG, 97064 Würzburg
Druck Vogel Druck & Medienservice GmbH, Leibnizstr. 5, 97204 Höchberg
Vertrieb MZV GmbH & Co. KG, 85716 Unterschleißheim
Internet: www.mzv.de
Kontakt Leserservice specials@chip.de

© 2017 by CHIP Communications GmbH. Nachdruck nur mit schriftlicher Genehmigung des Verlags.

Pressekontakt Dr. Petra Umlauf, pumlauf@chip.de, Tel. (089) 9250-4494
Bezugspreise Einzelheft: 9,95 Euro; Ausland: Österreich 11,50 Euro; Schweiz 19,50 SFr; BeNeLux 11,50 Euro
Nachbestellung (zzgl. Versand) chip-kiosk.de
Jahresabo (inkl. Versand) 54,80 Euro, Ausland: Österreich 69 Euro; Schweiz 117 SFr; BeNeLux 69 Euro
Abonnentenservice Abonnenten Service Center GmbH, CHIP-Aboservice, Postfach 225, 77649 Offenburg, Tel. (0781) 63 94 526 (Mo bis Fr, 8 bis 18 Uhr), Fax (0781) 84 61 91, E-Mail: abo@chip.de, kontakt@chip-kiosk.de

MagPi – das offizielle Raspberry Pi Magazin erscheint als Lizenzausgabe des MagPi Magazine der Raspberry Pi (Trading) Ltd., 30 Station Road, Cambridge, CB1 2JH. Alle Inhalte dieses Hefts unterliegen, sofern nicht anders gekennzeichnet, der Creative-Commons-Lizenz – Namensnennung – Nicht-kommerziell – Weitergabe unter gleichen Bedingungen 3.0 (CC BY-NC-SA 3.0).



> Leicht zu bauen und macht jede Menge Spaß

MIROBOT V2
FAHRBARER ROBOTER

Der Mirobot lässt sich ohne Werkzeug zusammenbauen – ist er also der perfekte Einstiegsroboter für aufstrebende Maker?

Roboter werden meist in einer sperrigen Verpackung geliefert, doch der Mirobot steckt in einer Papp-Schachtel, die kaum größer ist als ein Stapel CDs. Man denkt daher, dass er ebenfalls klein sein müsste, tatsächlich aber zeigt sich hier bereits das Talent des Schöpfers Ben Pirt: Die Verpackung des Mirobot wird zum Roboter.

Das Gehäuse des Mirobot besteht aus lasergeschnittenen Holz-Teilen, der Bausatz aus Brettchen, aus denen die einzelnen Teile nur noch herausgebrochen werden müssen. Für Motoren und Elektronik sind Löcher ausgeschnitten. Der ganze Bausatz wird dann sorgfältig zusammengesetzt und and den Ecken mit zwei Gummibändern zusammengehalten. Daher bleibt kaum Müll übrig. Die Holzreste der Brettchen sind nutzlos, lassen sich aber in der Biotonne oder im Holzofen entsorgen. Alles andere außer dem Pappkarton findet Verwendung, sogar die Gummibänder.

Der Zusammenbau ist einfach, Werkzeug wird nicht benötigt: Auch ohne Elektronik-Erfahrung kann man den farbig gedruckten Anweisungen samt Fotos leicht folgen, und hält nach 10 bis 20 Minuten einen funktionierenden Mirobot in Händen. Schwierigkeiten gibt es nur beim Stecken der unmarkierten Kabel auf die Hauptplatine, was auf bestimmte Weise getan werden muss, und dem Kalibrieren des Stifts mit dem beigelegten Werkzeug.

Einfacher Start

Der fertige Mirobot wird mit vier AA-Batterien betrieben und arbeitet vollständig unabhängig. Zunächst stellt das WLAN-Modul, das auf dem günstigen ESP8266-Chip basiert, einen offenen Hotspot bereit, an dem Sie sich per Smartphone, Tablet, Raspberry Pi oder Notebook anmelden können, um das Gerät zu konfigurieren.

Sie müssen ansonsten keinerlei Software installieren. Beim Mirobot erledigen Sie von Firmware-Updates bis zur Programmierung alles in Ihrem Webbrowser. Sie können ihn offline im offenen oder verschlüsselten Modus benutzen oder Ihre WLAN-Daten eintragen, um ihn in Ihr WLAN zu Hause – oder in der Schule – zu bringen.

Verbinden Sie den Mirobot mit dem WLAN, erhalten Sie Zugang zu den Mirobot-Apps von der offiziellen Website. Diese bieten eine breite Palette an Möglichkeiten, den Roboter zu programmieren: Etwa ein spartanisches, Block-basiertes Drag-and-Drop Interface, das lokal zur Verfügung steht, die Block-basierten Sprachen Blockly, Snap! und Scratch (auch

Tipp

ARCBOTICS SPARKI

Der Sparki wird mit einem On-board-Display und Sensoren geliefert, die der Mirobot nicht hat. Er ist daher insgesamt aber auch deutlich teurer.

ca. 150 €

arcbotics.com

mime.co.uk

ca. 70 €

Rechts Das Kit enthält das Gehäuse für den Roboter

wenn das aktuelle Scratch 2 nicht direkt auf dem Raspberry Pi läuft). Hinzu kommen Python und JavaScript samt Entwicklungsumgebung im Browser – inklusive Syntax-Highlighting – und eine einfache Zeichenfunktion. Es gibt auch eine Fernsteuerungs-Option. Hier können Sie den Mirobot mittels Schaltflächen auf dem Bildschirm steuern. Mit einem Smartphone wird der Mirobot so zu einem ferngesteuerten Auto – wenn auch zu einem ziemlich langsamen. Alle Funktionen lassen sich offline verwenden, sofern man eine Chrome-App installiert.

In der Praxis verhält sich der Mirobot wie ein typischer Turtle-Roboter. Er kann einen Stift heben oder senken und so auf Papier zeichnen, sich drehen, vor- und zurückfahren und seinen Lautsprecher piepsen lassen. Der überarbeitete Bausatz in Version 2.0 enthält einen optionalen Infrarot-Sensor, der den Mirobot zu einem Linienverfolger macht.

Mehr entdecken

Das englischsprachige Handbuch geht allerdings nicht so weit, wie Anfänger sich dies vielleicht wünschen würden. Insbesondere findet sich dort kein Hinweis auf den Einbau des Linienfolgers oder die Notwendigkeit, die Stiftposition und -höhe zu kalibrieren. Zudem fehlen Infos dazu, die Wegstrecke des Roboters durch Aufzeichnen und Messen einer Linie zu kalibrieren.

Zum Glück gibt es auf der offiziellen Website einen Bereich für Lernmaterialien, wo anschauliche Anleitungen für all diese Aufgaben enthalten sind – allerdings ebenfalls nur auf Englisch. Diese Anleitungen können sich sogar mit einem Mirobot in Ihrem Netz verbinden. Auf der Seite zur Kalibrierung der Motoren etwa können Sie per Knopfdruck eine Linie zeichnen lassen, und dann die von Ihnen gemessene Länge eingeben. Das Ergebnis wird an den Roboter zurückgeschickt und fest gespeichert. Sie finden auf der Seite noch viele weitere Tutorials und technische Dokumentationen.

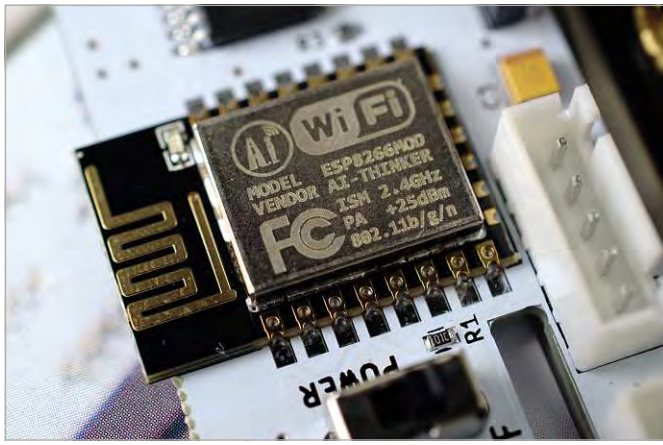
Das Fazit

Man muss den Mirobot einfach mögen. Vom einfachen Aufbau und der cleveren Verpackung bis zur Browser-basierten Schnittstelle macht das Gerät einfach Spaß und wird gerade bei jüngeren Makern groß rauskommen.

★★★★★

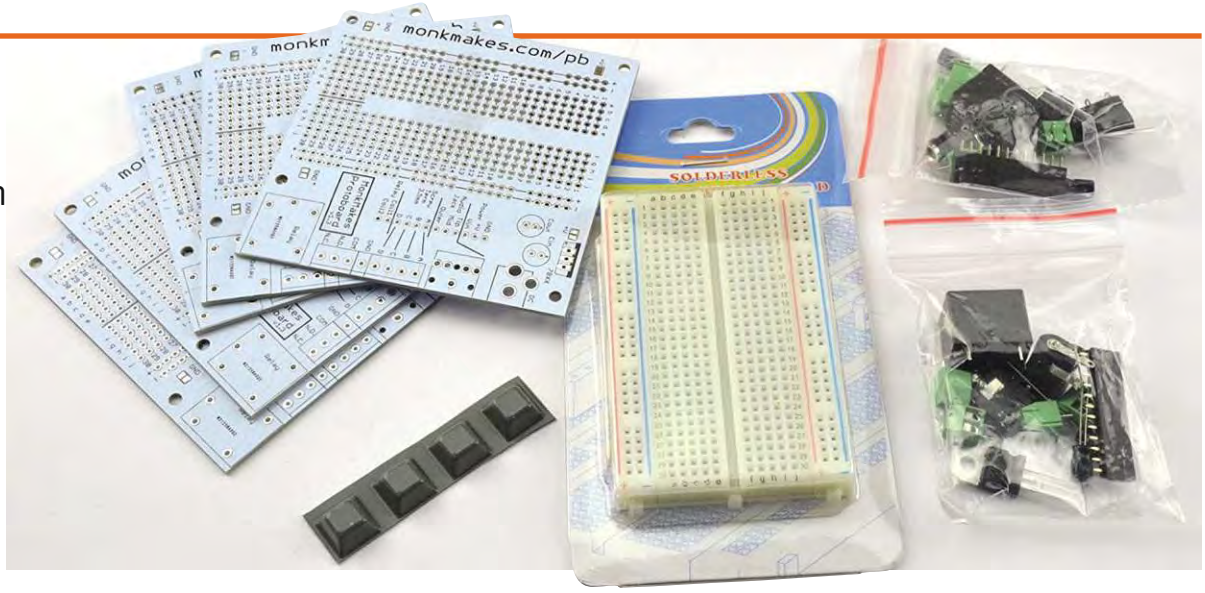
Oben Der Roboter lässt sich bequem per Browser programmieren

Unten Direkt auf dem Board ist ein WLAN-Modul verbaut



monkmakes.com/pb

ca. 30 €



> Macht den Bau von Einzelprojekten und Prototypen so einfach wie nur möglich

PROTOBOARD

Eine Steckplatine für den Bau von Prototypen – mit einigen Extras

Wie viele Hobbyelektroniker erstellt Simon Monk Prototypen gern auf einer Steckplatine, um sie zu testen und sicherzustellen, dass alles funktioniert wie gedacht. Mit herkömmlichen Prototypen-Platinen war er aber nach eigenen Angaben nie ganz zufrieden. Deshalb beschloss er, deren Schwächen zu beseitigen. Heraus kam das MonkMakes ProtoBoard. Nach einer erfolgreichen Kickstarter-Kampagne ist es nun bei Amazon UK als Deluxe Kit verfügbar (**amzn.eu/3YhwMaK**).

Das ProtoBoard wurde inspiriert durch die Perma-ProtoBoards von AdaFruit. Es übernimmt das Layout eines herkömmlichen Breadboards mit 400 Kontakten samt der üblichen Kontakt- und Verteilerleisten, wenn auch mit längeren Leisten an der Unterkante des Boards, sodass sich Module wie ESP01 oder NRF24 mit 2×4 oder 2×5 Kontakten montieren lassen – eine gute Idee.

Der auffälligste Unterschied betrifft jedoch den hinzugekommenen Bereich auf der rechten Seite. Dort lassen sich zusätzliche Komponenten aufbauen, die nicht direkt auf ein Breadboard passen. Dazu gehören unter anderem viele Relais, 3,5 mm Audio-Buchsen und die universellen Schraubklemmen. An der Oberkante gibt es noch einen Bereich, wo man eine Stromversorgung aus einer Hohlstecker-Buchse, einen linearen Spannungsregler oder Kondensatoren aufbauen kann. Diese kann man mit den Stromschienen auf dem Board verbinden, indem man die vorbereiteten +V- und GND-Brücken schließt.

In dem bei Amazon erhältlichen Deluxe-Kit sind zwei Beutel mit Bauteilen sowie ein lötfreies Breadboard halber Größe und vier Gummifüße enthalten. Zum Entwickeln von Prototypen empfiehlt Simon, das Breadboard fest auf eines der Protoboards zu löten und das Seitenteil vollständig zu bestücken. Wenn der Schaltkreis dann ordnungsgemäß funktioniert, ist es ein Leichtes, die Komponenten vom Breadboard und der Seitenfläche auf ein zweites ProtoBoard zu übertragen (bzw. zu kopieren). Das ist eine tolle Idee, die den Aufbau deutlich erleichtert.

Das ProtoBoard eignet sich nicht nur für beliebige Projekte, die auf ein Breadboard mit 400 Kontakten passen, sondern auch für den Einsatz mit Mikrocontrollern wie Particle Photon oder Arduino Pro Mini – oder zum Aufbau eines eigenen Arduino.

Tipp

ADAFRUIT PERMA-PROTO

Diese in drei Größen erhältliche Prototypen-Platine entspricht dem klassischen Breadboard, aber ohne die zusätzlichen Anschlüsse des MonkMakes ProtoBoard.

ca. 13 €

bit.ly/2jWCthM

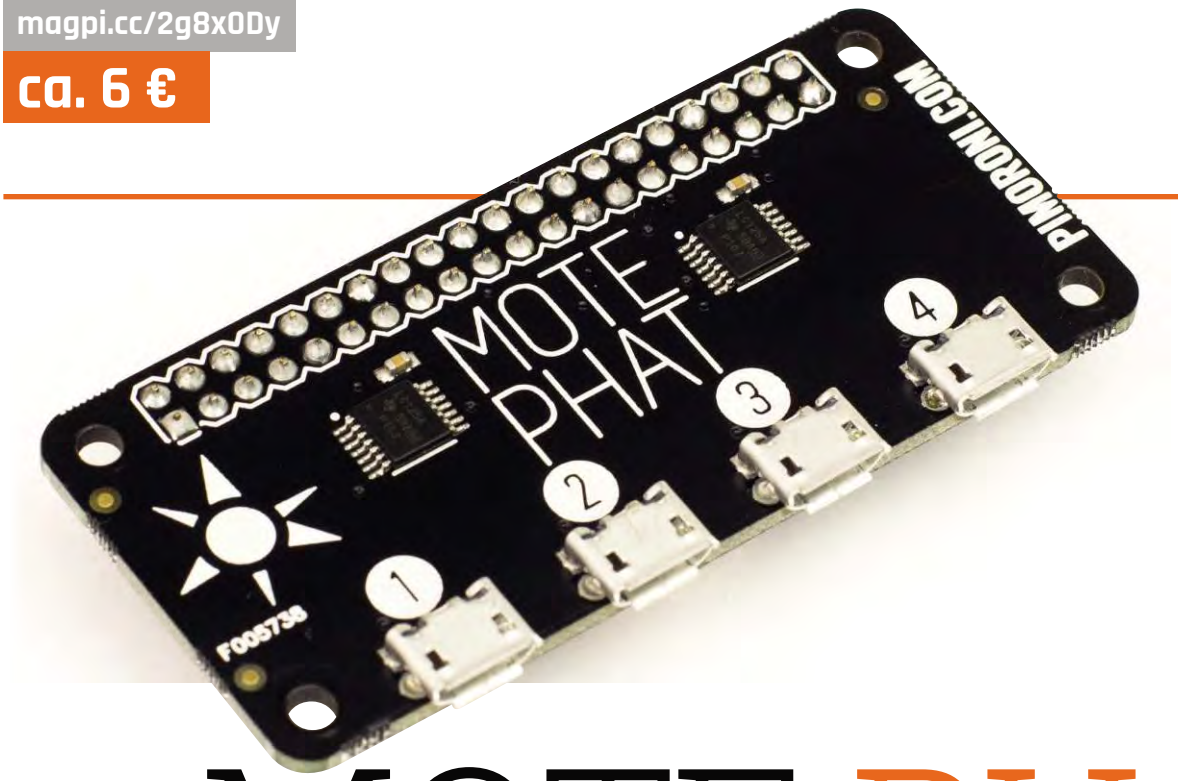
Das Fazit

Mit seinem cleveren Design und den durchkontaktierten Bohrungen vereinfacht das ProtoBoard das Übertragen von Projekten von einer Steckplatine auf ein Prototypen-Board. Vor allem das Seitenteil ist sehr praktisch, wenn Bauteile verwendet werden, die sich nicht direkt auf ein Breadboard stecken lassen.

★★★★☆

magpi.cc/2g8x0Dy

ca. 6 €



Steuern Sie Ihre Mote LED-Sticks direkt vom Raspberry Pi aus

MOTE PHAT

Eine kompaktere Lösung, um Mote LED-Sticks zu steuern

Mit 16 superhellen RGB-LEDs sind die Mote LED-Sticks von Pimoroni (ca. 10 €) ideal für die Montage unter Schränken – etwa als Mood Light. Auch für beeindruckende Effekte lassen sie sich verwenden. Um die Motes anzusteuern, war bislang ein Mote Host erforderlich, der sich per USB mit beliebigen Computern – auch einem Pi – verbinden lässt. Der Mote pHAT leistet das Gleiche, ist aber kompakter und leichter handzuhaben. Die pHAT-Platine passt perfekt auf den Pi Zero, sodass Sie diese unauffällig unterbringen können. Zunächst müssen Sie die GPIO-Buchsenleiste auf den Mote pHAT löten. Dabei können Sie eine Stapelleiste verwenden und das Board etwa zusammen mit anderen Boards verwenden. So könnten Sie beispielsweise Mote-Sticks dazu einsetzen, um die Sensordaten eines Enviro pHAT anzuzeigen, oder sie mit Tastern oder Schaltern zu steuern.

Wie der Mote Host besitzt der Mote pHAT microUSB-Buchsen für die vier Ausgangskanäle, sodass sich bis zu vier Mote Sticks ansteuern lassen. Auch eine Python Bibliothek (**magpi.cc/2fw4oFX**) mit ein paar Beispielen stellt der Hersteller zur Verfügung. Darunter sind ein beeindruckender Regenbogen-Effekt, Farbwechsel-Effekte und ein CheerLights-Programm zur Steuerung via Twitter. Die Bibliothek ist einfach zu verwenden und erlaubt es, jeden RGB-Pixel separat zu steuern. Sie triggern sie mit einem `show` Befehl; mit `clear` schalten Sie sie wieder ab. Mit einer `for` Schleife haben wir schon nach kurzer Zeit einen einfachen Verfolger-Effekt realisiert. Da die LEDs dem APA102-Standard („DotStar“) entsprechen und eine hohe Datenrate unterstützen, reagieren sie fast verzögerungsfrei. Es kommt nicht zu Timing-Problemen, wie dies bei NeoPixels geschehen könnte. Mittels eines Flask API können Sie diese sogar von Ihrem Handy oder einem Webbrowser aus steuern. Das englischsprachige Tutorial von Pimoroni (**magpi.cc/2g9g143**) zeigt, wie man die LEDs per Homekit mit einem iPhone steuert und per Siri-Sprachkommandos ein- und ausschaltet. Das ist cool, um Freunde zu beeindrucken! Mit Google Assistant oder Alexa sollte dies jedoch auch möglich sein.

Tipp

MOTE HOST

Der ältere Mote Controller wird mit dem USB-Port eines Computers, etwa eines Pi 3, verbunden. Wie der pHAT besitzt er vier microUSB-Ports.

ca. 14 €

magpi.cc/2g9iyLs

Das Fazit

Die Mote Sticks sind toll, noch besser als NeoPixels. Zudem bietet der neue Mote pHAT eine kompakte, unauffällige Methode, sie ohne Anschluss an den USB-Port eines Notebooks oder Pis zu steuern. Er ist seinen Preis von rund 6 € absolut wert. Nur die Sticks und Kabel müssen Sie noch dazukaufen.

★★★★☆

> Die schlankeste Kodi-Version aller Zeiten

LIBREELEC 7

LibreELEC ist ein neues Media-Center-OS für den Raspberry Pi, das auf eine bewährte Basis setzt. Aber ist es besser als OpenELEC?

Es ist ein Weilchen her, dass wir uns ein Media Center für den Raspberry Pi angesehen haben. Der „Markt" wird schon seit einer Weile von OSMC und OpenELEC beherrscht. Daher gab es auch kein großes Bedürfnis nach Alternativen. Beide sind sehr gut und bieten bis auf Kleinigkeiten dasselbe. Wollte man also nicht das Rad neu erfinden, gab es da nicht viel hinzuzufügen.

Womit wir zu LibreELEC kommen. Es bietet nichts wesentlich Neues, aber es ist noch schlanker als OpenELEC. Es dürfte sich um die schlankste Kodi-Version (mit XBMC-Wurzeln) handeln, die Sie finden werden. Das verwundert auch nicht weiter, wenn man weiß, dass LibreELEC als Ableger von OpenELEC begann. Falls Sie sich mit Open-Source-Software auskennen: Es ist ein bisschen wie bei OpenOffice und LibreOffice.

LibreELEC versucht den offeneren, freien Geist früherer OpenELEC-Versionen weiterzuführen und dennoch die allerneuesten Entwicklungen von Kodi umzusetzen. Erfreulicherweise soll auch die Stabilität nicht zu kurz kommen. Auf der Website kann man diese Vorsätze nachlesen. Nach Beta-Versionen für die neue Version 17 von Kodi folgten stets kurz danach Pre-Releases der nächsten LibreELEC-Version.

Bei diesem Test konzentrieren wir uns auf die bei Redaktionsschluss aktuelle Beta-Version LibreELEC 7.95. Installer sind für alle wichtigen Betriebssysteme vorhanden (Windows auf Heft-DVD, Mac OS, Linux), sodass die Einrichtung kinderleicht ist. Der Installer lädt ein Image herunter (Stable, Beta muss von Hand geladen werden), für das Sie ein Zielverzeichnis auswählen. Anschließend können Sie es auf die SD-Karte schreiben. Das dauerte bei uns im Test nur rund eine Minute. Insgesamt ist der Installer sehr komfortabel und selbsterklärend. Es gibt sogar einen Fortschrittsbalken, um den Vorgang verfolgen zu können.

LibreELEC im Praxistest

Beim Booten sieht alles vertraut aus. Vor dem Reboot ins Media Center wird die SD-Karte angepasst, um den verfügbaren Platz auszunutzen, und ein paar weitere Aufgaben werden ausgeführt.

Ab hier dürfte Ihnen alles vertraut vorkommen. LibreELEC enthält die Standardversion von Kodi ohne Custom Skins, wie etwa bei OSMC. Nun werden noch einige Details abgefragt, und schon können Sie sich mit dem integrierten WLAN-Modul des Raspberry Pi 3 am WLAN anmelden. Außerdem kommt die Software mit einer guten Auswahl an Treibern für WLAN-Module, falls Sie einen Pi 2 verwenden. Damit ist die Einrichtung beendet und Sie können anfangen, Medienquellen hinzuzufügen. Was uns besonders gefal-

Tipp

OSMC

OSMC ersetzt das Standard-Interface von Kodi mit seinem eigenen. Dadurch fühlt es sich etwas anders, aber immer noch solide an.

KOSTENLOS

osmc.tv

KOSTENLOS

Oben Selbst die Beta-Version bietet schon viel Komfort

len hat bei OpenELEC: Dank der Kodi-Umgebung, mit der viele vertraut sind, geht dieser Vorgang schnell und einfach vor sich.

Deutschsprachige Nutzer sollten das deutsche Sprachpaket nachinstallieren. Dies geschieht über **Add-ons | Download | Look & Feel | Languages**. Wechseln Sie anschließend zu dieser neuen Sprache – LibreELEC spricht von nun an Deutsch. Jetzt müssen Sie nur noch die Tastaturbelegung umstellen: Dies geschieht über **System | Benutzeroberfläche | Regional**. Sie sollten dabei nicht nur die deutsche Tastaturbelegung anwählen, sondern die englische auch abwählen. Das System ist selbst in der Betaversion stabil. Es ruckelt nichts, die Medienwiedergabe arbeitet einwandfrei und brach bei unseren Tests kein einziges Mal ab. Weder bei Funktionen noch bei der Benutzerfreundlichkeit müssen Sie Abstriche machen.

Es handelt sich allerdings nicht um eine revolutionäre Neuerung. OpenELEC und OSMC laufen auf Pi 2 und Pi 3 einwandfrei. Falls Sie OpenELEC installiert haben und keine Probleme haben, lohnt es sich eigentlich nicht, neu zu installieren und die ganzen Medienquellen erneut hinzuzufügen.

> „Wir finden, dass LibreELEC einen Tick besser für den Raspberry Pi optimiert ist"

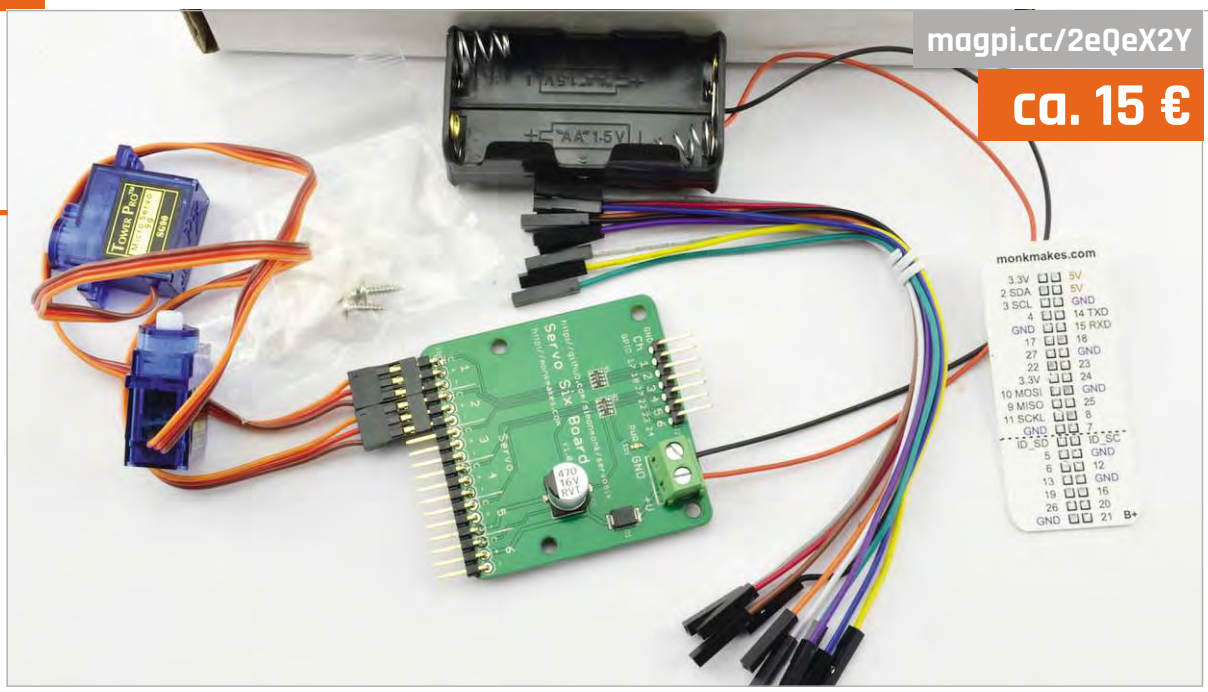
Achtung: Auch wenn LibreELEC Raspberry Pis der ersten Generation (und den Pi Zero) unterstützt, ist die Performance bei diesen schwachbrüstigen Modellen etwas eingeschränkt. Ansonsten können wir LibreELEC uneingeschränkt empfehlen. Selbst die Betaversion macht bereits einen guten Eindruck. Wer es sicherer mag, wählt jedoch lieber „Stable".

Mitte Das Installations-Tool ist leicht zu bedienen

Das Fazit

Wenn Sie Kodi unmodifiziert und turbo-schnell mögen, ist LibreELEC das Richtige für Sie. Doch falls Ihr Media Center gut läuft, lohnt sich der Umstieg nicht unbedingt.

★★★★★



> Mit dem Kit lassen sich sechs Servos steuern

SERVO KIT

Ein Kit für einfache Robotik-Projekte mit Ihrem Raspberry Pi

Platinen zur Ansteuerung von Motoren präsentieren wir in der MagPi recht häufig, Servo-Controller aber nur selten. In gewisser Weise sind sie sich ähnlich: Beide treiben einen Elektromotor an, der von einem Pi gesteuert wird. Der Hauptunterschied besteht in der Leistung und Genauigkeit der Bauteile. Motoren laufen für gewöhnlich einfach, während man einem Servo eine bestimmte Drehung vorgeben kann. Die Einschränkungen beider Typen lassen sich mit Code und Know-how überwinden, aber es ist immer besser, den jeweils vorgesehenen Motortyp einzusetzen.

Das Servo Kit von Monk Makes besteht aus mehreren Teilen, unter denen das Servo Six Board, ein Servo-Controller, heraussticht. Der Rest des Kits besteht aus zwei einfachen Servos, einigen Jumperkabeln und einem Batteriehalter. Das reicht, um mit Servos loszulegen.

Die Platine schließen Sie über sieben Kabel (6 x GPIO, 1 x GND) an den Raspberry Pi an. Da es so wenige Verbindungen gibt, arbeitet die Platine nicht als HAT. Stattdessen schließen Sie sie über die Jumperkabel an. Für viele Projekte ist das praktisch, denn die restlichen GPIO-Pins bleiben dadurch für andere Zwecke frei. Falls ein HAT zu Ihrem Projekt gehört, müssen Sie sich einen Workaround überlegen.

Der Aufbau ist einfach, weil lötfrei, und auf der Servo-Six-Seite (**magpi.cc/2eXxxtr**) sowie im GitHub-Repository stehen jede Menge Infos, mit denen Sie einen einfachen Schaltkreis aufbauen und programmieren können. Der Code ist wie üblich in Python geschrieben, nur werden hier Winkel benutzt statt der üblichen Kommandos (Vorwärts, Links, Rechts, Stop), wie man sie von anderen Monk Makes Boards kennt.

Gibt man einen Winkel vor, dreht der Servo auf die Stellung, die er für diesen Winkel hält. Für Ihr Projekt werden Sie ihn jedoch kalibrieren oder zumindest herumprobieren müssen. Außerdem sollten Sie verstehen, welcher Winkel welche Drehung verursacht, wenn Sie den Servo „zurück“ fahren lassen möchten. Darüber hinaus gibt es ein paar kleine weitere Tricks, die Sie manchmal benötigen, um den Servo in die richtige Richtung drehen zu lassen.

Das Servo Six Board selbst ist jedoch toll. Sie können damit sechs unabhängige Servos steuern, und im Code ganz einfach anhand der Nummern auseinanderhalten. Da nur zwei Servos dabei sind, werden Sie für umfangreichere Projekte zusätzliche Servos benötigen.

Das Fazit

Ein tolles Set, um zu lernen, wie man Servos verwendet. Selbst für etwas größere Projekte lässt es sich gut verwenden. Auch der Preis stimmt.

★★★★☆

Tipp

ADAFRUIT MINI PAN-TILT KIT

Ein Servo-Kit für bestimmte Fälle, das sich aber für andere Zwecke verwenden lässt.

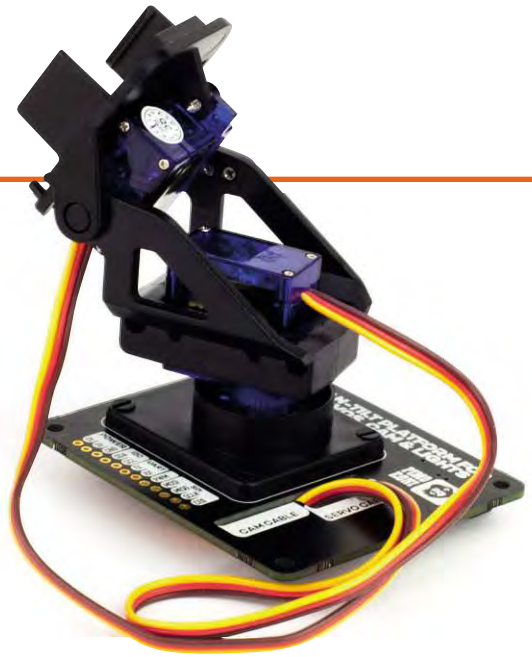
ca. 18 €

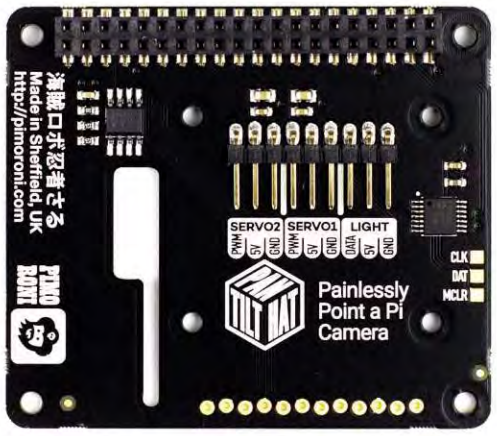
bit.ly/2jD95AB

bit.ly/2jPgLz8

ca. 35 €

Programm-Code auf Heft-DVD





> Sehr praktisch, um etwa Räume zu überwachen

PAN-TILT HAT

Ein horizontal und vertikal schwenkbarer Halter für die RasPi-Kamera

Die Raspberry-Pi-Kamera ist eines der besten Zubehörteile, die es gibt, denn mit ihr kommen Sie günstig zu einer Bilderfassung für den Pi. Nur bleibt sie nicht von selbst aufrecht stehen – Sie brauchen einen Halter.

Der Pan-Tilt HAT ist ein Halter, kann aber noch viel mehr. Das Kameramodul wird auf das Ende eines beweglichen Arms montiert, der auf dem HAT sitzt. Über zwei Servos und die horizontalen und vertikalen Gelenke kann die Kamera exakt gesteuert werden.

Fertig aufgebaut sieht das Ganze äußerst putzig aus und verleiht Ihrem RasPi sofort Persönlichkeit, wenn es sich im Raum umsieht.

Nützlich ist das Ganze auch. Sie können den Pan-Tilt HAT so einrichten, dass er ein Zimmer überwacht, und die Kameraposition dann per VNC oder SSH steuern.

Alternativ lässt sich auch ein Raspberry Pi mit Gesichtserkennungssoftware ausrüsten und mit dem Pan-Tilt HAT verbinden. Pimoroni, der Hersteller, empfiehlt sogar den Einsatz als Roboterauge.

Einrichtung

Zunächst müssen Sie den HAT einrichten. Dafür gibt es zum Glück eine (englischsprachige) Anleitung online (**magpi.cc/2hR4NFC**).

Das Board besitzt eine GPIO-Buchsenleiste sowie Anschlüsse für die Servos. Die beiden Kabel, die zum Arm führen, werden mit den Anschlüssen Servo1 (Schwenken) und Servo2 (Neigen) auf dem Board verbunden. Ein dritter Servo-Kanal kann verwendet werden, um einen optionalen NeoPixel-Streifen als Beleuchtung anzusteuern.

Kamerasteuerung

Den Code können Sie sich von unserer **Heft-DVD** oder bei GitHub (**magpi.cc/2hRrjyo**) herunterladen. Um den HAT zu steuern, importieren Sie das **`pantilthat`**-Modul in Python. Danach können Sie die Kameraposition mit den Methoden **`pan()`** und **`tilt()`** steuern. Diese akzeptieren Werte zwischen -90 und 90. Um die Kamera geradeaus nach vorne zu richten, verwenden Sie diese Kommandos:

```
pantilthat.pan(0)
pantilthat.tilt(0)
```

45 Grad nach oben geht so:

```
pantilthat.tilt(-45)
```

Um die Kamera ganz nach links schauen zu lassen:

```
pantilthat.pan(90)
```

Wir hätten gern noch mehr Beispiele gehabt, etwa für das Aufnehmen mit der Kamera oder eine Gesichtsverfolgung. Aber auch so hatten wir Spaß mit dem Pan-Tilt HAT und freuen uns auf weitere Experimente.

Das Fazit

Mit diesem putzigen, vielseitigen Zubehör können Sie eine Menge Spaß haben. Nur müssen Sie erst noch ein bisschen recherchieren.

★★★★☆

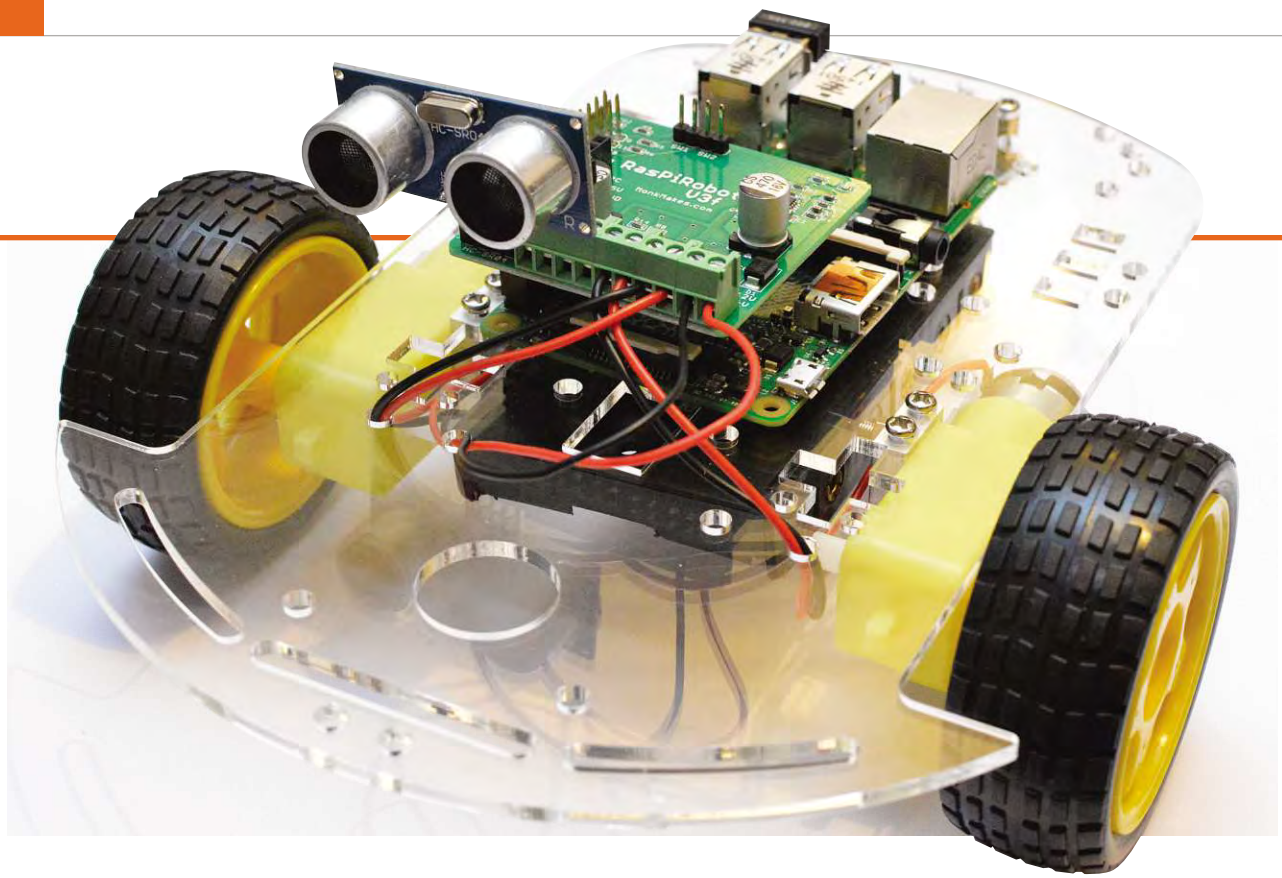
Tipp

RASPBERRY PI CAMERA MOUNT

Sie bewegt sich nicht, aber wenn Sie nur eine Kamerahalterung suchen, bekommen Sie diese hier günstig bei Pi Hut.

ca. 4 €

magpi.cc/2hRtkdT



> Bauen Sie einen Pi-Rover – es fehlen nur der RasPi und die Akkus

RASPIROBOT ROVER KIT

Tipp

GOPIGO

Beim GoPiGo handelt es sich um einen weiteren einfachen Roboter-Bausatz. Auch dieser lässt sich für viele Zwecke verwenden.

ca. 100 €

magpi.cc/2hA8f6i

Hier haben wir einen Bausatz, der mehr sein möchte als nur ein Roboter. Doch eignet er sich, um Einsteigern das Programmieren beizubringen?

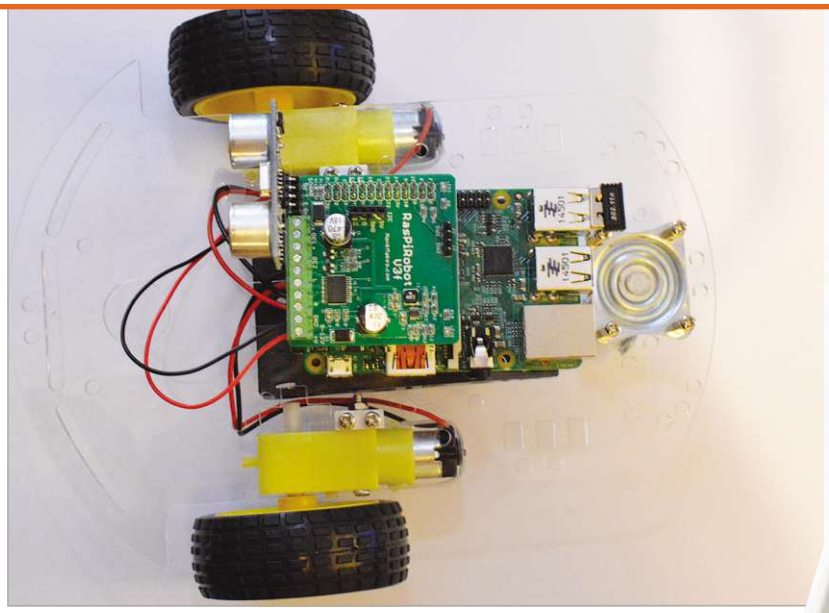
Bausätze von Monk Makes gefallen uns eigentlich immer. Sie bieten gute Funktionen, eine tolle Dokumentation und sie sind meist preislich recht attraktiv. Auch das RasPiRobot-Board hat im Test kürzlich gut abgeschnitten. Daher freuen wir uns, es in einem Pi-Roboter-Bausatz zu sehen, so wie Simon Monk sich das gedacht hat.

Dieser Roboter-Bausatz besteht eigentlich aus zwei Kits. Neben einem einfachen, programmierbaren Roboter finden Sie einen Drucktaster und eine Raspberry Squid LED. Diese funktionieren auf dem RasPiRobot-Board, auch wenn es nicht als Roboter aufgebaut ist. Sie können diese auch erst dann anschließen, nachdem der Roboter fertig gebaut ist. Das ist eine interessante Idee, denn theoretisch können Sie Taster und LED zum Einstieg in die GPIO-Programmierung verwenden und anschließend den Roboter aufbauen und so noch andere Dinge basteln.

Der Bau des Roboters selbst geht schnell und einfach von der Hand. Die Grundplatte besteht aus Acrylglas, der Antrieb aus zwei Motoren. Am Heck ist eine Lenkrolle angebracht, der Batteriehalter kann an der Unterseite montiert werden. Der Pi wird einfach mit Uhu Patafix oder doppelseitigem Klebeband aufgeklebt. Die restlichen Bauteile passen genau auf den Motor-Controller, der auf den Pi aufgesteckt wird. Das ist ganz einfach. Es gibt einige Video-Tutorials von Simon, die zeigen, wie es funktioniert. Eine Schritt-für-Schritt-Anleitung wäre allerdings trotzdem nett. Der Zusammenbau war in ca. 50 Minuten erledigt. In dieser Zeit ist aber auch das schwierige Abziehen der

magpi.cc/2i7kJzh

ca. 50 €



Oben Bei diesem Bausatz sind Sie schnell fertig, es gibt nicht viel zu basteln. Aber dafür sind viele Extra-Teile beigelegt

Schutzfolie von der Grundplatte und das Herauskramen von passenden Akkus enthalten.

Der fertige Roboter ist recht robust. Die Grundplatte ist klein und biegt sich nicht durch. Der RasPi läßt sich dort so anbringen, dass er nicht herunterfällt. Im Bausatz sind Timer-Scheiben enthalten, die mit zusätzlichen Sensoren zu Geschwindigkeitssensoren mutieren. Praktischerweise sind an der Grundplatte diverse Punkte vorhanden, wo man diese befestigen kann. Die Vorderseite eignet sich besonders für Linien- und Näherungssensoren oder einen schwenk- und neigbaren Arm. Sie müssen allerdings selbst recherchieren, um herauszufinden, wie Sie diesen anschließen.

Die Programmierung ist mittels der Monk Makes Bibliotheken recht einfach. Das Anleitungs-Heft, das dem Roboter beiliegt, enthält ein paar kurze Beschreibungen, wie man LED und Taster zum Laufen bekommt und den Roboter für Fernbedienung oder Autonomie programmiert. Sie finden dort auch etwas über den Ultraschall-Abstandsmesser, sodass Sie mit dessen Programmierung beginnen können. All diese Teile lassen sich miteinander kombinieren, um verschiedene Sachen zu machen.

> „Taster und LED können Sie auch zum Einstieg in die GPIO-Programmierung verwenden“

So könnte etwa der Taster eine kleine Mission starten, bei der der Roboter auf eine Wand zufährt und anhand der Daten des Näherungssensors die LED-Farbe beim Näherkommen ändert. Auch wenn dies nur eine kleine Modifikation darstellt, eröffnet sich so eine ganze Reihe weiterer, einfacher Anpassungen und Spielereien.

Der RasPiRobot Rover gefällt uns gut. Wir haben bereits viele ähnliche Roboter gebaut. Doch auch wenn uns diese immer Spaß gemacht haben, bieten sie meist nicht den Zusatznutzen dieses Kits. Außerdem können wir unseren Pi hier problemlos ausbauen und müssen uns nicht mit kleinen Schräubchen herumplagen. Und schließlich ist dies für seinen Preis ein tolles Einsteiger-Kit für Kinder, die sich für den RasPi und für Elektronik interessieren.

Das Fazit

Der RasPiRobot Rover ist ein schöner Bausatz und bietet einen tollen Einstieg in die Robotik, richtet sich aber eher an Fortgeschrittene.

★★★★☆

> Es ist zu hoffen, dass die Box bald auch mit dem Pi 3 kooperiert

NEXTCLOUD BOX

Kann Ihnen ein Barebone-Gehäuse, in das Sie Ihren Pi einbauen, zu Datensicherheit und Unabhängigkeit verhelfen?

Durch ihren niedrigen Stromverbrauch sind die Raspberry Pis gut geeignet für Server-Anwendungen, die rund um die Uhr laufen. Ein NAS-Gerät ist eine mögliche Anwendung. Doch wer hier seine eigene Lösung bastelt, muss mit externen USB-Festplatten und diversen Netzteilen hantieren – das ist wenig erquicklich.

Hier kommt die Nextcloud Box ins Spiel. Sie entstammt einer Partnerschaft zwischen den WD Labs des Speicherriesen Western Digital und dem Open-Source-Projekt Nextcloud. Mit dem PiDrive von WD wird die Nextcloud Box zu einem Server, auf dem eine Variante der Nextcloud-Software läuft.

Beim Auspacken stellt man schnell fest, dass kein Pi dabei ist. Die Nextcloud Box enthält eine 1 TByte große USB-Festplatte, ein cleveres Gehäuse mit Magnetverschluss, ein Stromkabel mit verschiedenen Enden, ein Datenkabel, Schrauben zur Befestigung, einen Schraubendreher, eine 4 GB große microSD-Karte sowie ein Netzteil, das Pi und Festplatte gemeinsam versorgt. Der Pi muss vom Anwender kommen. Zum Zeitpunkt des Tests wurde nur der Pi 2 unterstützt, Kompatibilität zum Pi 3 soll aber mit einem kostenlosen Software-Update nachgereicht werden.

Cloud-Setup

Die Installation geht einfach vonstatten: Befestigen Sie Ihren Raspberry Pi mit den beiliegenden Schrauben im Gehäuse, stecken Sie die Kabel ein und stellen Sie ein Netzwerkkabel bereit. Die 4 GByte große microSD-Karte wird nur für den ersten Boot benötigt. Dann wird ganz automatisch das Betriebssystem Ubuntu Core auf die 1 TByte große Festplatte kopiert, was ungefähr zehn Minuten dauert. In der Praxis hat uns überrascht, dass man Display oder Tastatur eigentlich gar nicht braucht. Die gesamte Administration erfolgt über eine Weboberfläche – mit Ausnahme einiger komplexer Prozeduren, die über SSH laufen müssen.

Ab hier läuft die Verwendung der Nextcloud Box wie die jeder anderen Installation von Nextcloud: Userkonten, Verschlüsselungs-Einstellungen und zusätzliche Features können direkt im Browser eingerichtet werden.

Standardmäßig ist Ihre Nextcloud Box nur aus Ihrem lokalen Netzwerk erreichbar, entweder per Browser-Oberfläche oder spezieller Client-Software. Das Umkonfigurieren für den Zugang von außen gehört zu den wenigen Dingen, die für technisch Nicht-Versierte schwierig sind: Dazu müssen Sie sich nämlich per SSH einloggen, einen DynDNS-Dienst konfigurieren, ein Skript ausführen, das ein kryptographisches Zertifikat von Let's Encrypt herunterlädt und schließlich noch Port 443 am Router für die Next-

Tipp

WD LABS PIDRIVE BERRYBOOT EDITION

Falls Sie die Nextcloud Software nicht brauchen, können Sie PiDrive auch separat von WD Labs beziehen. Gehäuse und Netzteil müssen Sie dann aber zusätzlich erwerben.

ca. 60 €

wdlabs.wd.com

magpi.cc/2ivwjou

ca. 70 €

cloud Box öffnen. Wenn Sie alles geschafft haben, erhalten Sie verschlüsselten Zugriff auf Ihre Daten – einschließlich Zwei-Wege-Synchronisation. Dies stellt eine gute Möglichkeit dar, die Fotos und Videos Ihres Handys zu sichern – von überall auf der Welt und zum Nulltarif.

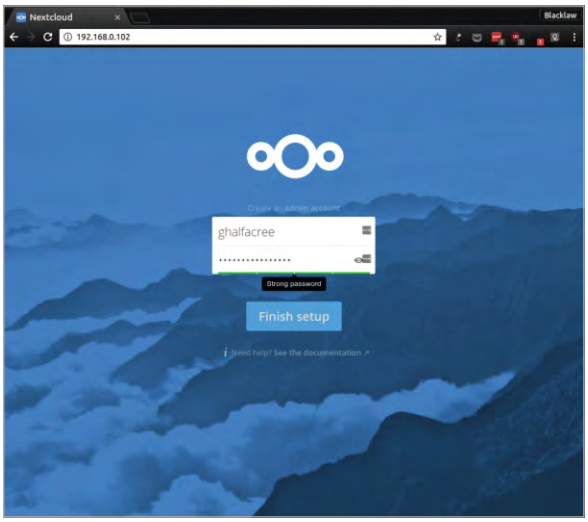
Das Arbeitstempo ist überraschend flüssig: Obwohl sich Netzwerk-Anschluss und Festplatte den einzigen USB-Kanal des Pi teilen, ging das Kopieren von Daten auf die Nextcloud Box zügig und ohne nervige Pausen oder Ruckler voran.

Falls Sie keine Lust haben, einen Pi nur für die Nextcloud zu reservieren, gibt es zum Herumspielen noch das darunterliegende Betriebssystem Ubuntu Core. Nextcloud wird als „Snap“ installiert, also als Container mit allen Abhängigkeiten, die Sie benötigen, um die Software einzusetzen. Darüber hinaus können Sie so viele Snaps installieren, wie Sie möchten und Features hinzufügen, die das, was Sie im Nextcloud App-Store finden, weit übertreffen. Der Nextcloud Snap wird dadurch dank des speziellen Konzepts zu keiner Zeit gefährdet.

Leider läuft aber nicht alles an der Nextcloud Box so rund wie die Software. Das Gehäuse mit dem Nextcloud-Branding macht es erforderlich, dass die Kabel um einige enge Ecken verlegt werden, bevor sie durch Aussparungen nach draußen geführt werden. Das größte Manko stellt aber das Stromkabel für Pi und Festplatte dar. Es wird aus dem Gehäuse herausgeführt, um 180 Grad gedreht und wieder hineingeführt, damit es den Micro-USB-Port des Pi erreicht.

Doch wenn man die Nextcloud Box nicht gerade ständig ein- und aussteckt, kann man das System nach dem Setup mehr oder weniger vergessen: Kabel verlegen, Software konfigurieren – von da ab aktualisiert sich die Box automatisch über Ubuntu Cores Update-Mechanismus.

> „Standardmäßig ist die Nextcloud Box nur aus Ihrem lokalen Netzwerk erreichbar“



Oben Auf der Nextcloud Box läuft Ubuntu

Das Fazit

Nextcloud ist leistungsfähig, flexibel und lässt sich sehr genau konfigurieren. Kurz: eine tolle Software, die perfekt auf dem Raspberry Pi 2 läuft. Die Hardware von WD Labs sollte aber noch einmal überarbeitet werden, denn die Kabel werden im Inneren der Box bedenklich abgeknickt.

★★★★☆

PRAKTISCHE EDUKITS

Jumperkabel, LEDs, Breadboard und vieles mehr: Für die ersten Bastelprojekte ist so manches Zubehör nötig. Starterkits erleichtern Ihnen den Einstieg, sind dafür aber etwas teurer als die Einzelteile.

CAMJAM EDUKIT #1

Hersteller: The Pi Hut
Dokumentation: Englisch
Preis: ab € 12,89
Info: amzn.to/2liX6Jd

Die CamJam Edukits wurden für die Raspberry Pi Jams in Zusammenarbeit mit The Pi Hut entwickelt. Sie sind sehr klein und in Deutschland verhältnismäßig teuer. Das EduKit #1 enthält 3 LEDs, einen Taster, einen Buzzer und ein Steckbrett, dazu vier Transistoren und Jumperkabel.

PROJECT SUPER STARTER KIT FOR RASPBERRY PI

Hersteller: Sunfounder
Dokumentation: Englisch
Preis: € 44,44
Info: amzn.to/2lPDEQX

Der amerkanische Anbieter hat ein großes Sortiment an Kits für den Raspberry Pi und für Arduino, die man über Amazon kaufen kann. Das Super Kit enthält sämtliches Zubehör, das Sie für die 15 Projekte benötigen, die im Handbuch beschrieben sind.

DAS FRANZIS RASPBERRY PI LERNPAKET

Hersteller: Franzis
Dokumentation: Deutsch
Preis: € 49,95
Info: bit.ly/2lKe3gp

Auch Franzis bietet eine Reihe von Kits. Ihr Vorteil: Die Beschreibungen sind auf Deutsch. Das Lernpakt enthält 41 Bauteile und ein Handbuch mit 192 Seiten. Unter den 20 beschriebenen Projekten sind unter anderem eine Fußgängerampel und eine Weltzeituhr.

ROBOTER-AUTOS MIT DEM RASPBERRY PI

Autor: Ingmar Stapel
Verlag: Rheinwerk
Preis: € 29,90
ISBN: 978-3-8362-4294-3
Info: bit.ly/2kDldQj

Ein eigenes, autonom fahrendes Auto, das Hindernissen selbstständig ausweichen kann – wer so ein Projekt meistern will, ist mit dem Buch von Ingmar Stapel bestens beraten. Auch Einsteiger ohne Vorkenntnisse finden hier alles Nötige von den Grundlagen bis hin zu kompletten Einkaufslisten.

Dazu ist das Thema in zwei große Blöcke unterteilt. Im ersten Teil geht es darum, ein einfaches Auto zu bauen, das ferngesteuert wird. Dabei werden die wichtigsten Grundlagen zum Aufbau und zu den Komponenten, aber auch Handwerkliches wie etwa das Löten erklärt. Außerdem gibt es eine kurze Einführung zum Programmieren mit Scratch. Im zweiten Teil kann man sein gewonnenes Wissen erweitern. Es werden die einzelnen Sensoren im Detail erklärt; hinzu kommt eine Einführung in die Programmierung in Python. Das Buch ist eine Fundgrube für viele praktische Tipps zu allen Arbeitsschritten eines solchen Autobaus.

Praktisch: Den Code zum Download, verlinkte Einkaufslisten und vieles mehr werden zusätzlich online im Blog des Autors zur Verfügung gestellt. Entsprechende Angaben finden Sie jeweils an geeigneter Stelle im Buch.

Fazit ★★★★★

HOMEAUTOMATION MIT DEM RASPBERRY PI

Autor: E. F. Engelhardt
Verlag: Franzis
Preis: € 30,00
ISBN: 978-3-645-60391-1
Info: bit.ly/2kFXtdE

Hier geht's ans Eingemachte. Das Buch von Pi-Experte Engelhardt ist nicht für das kleine Projekt zwischendurch geeignet. Stattdessen bekommen Sie geballtes Wissen auf mehr als 350 Seiten in hoher Qualität. Das ist nicht immer ganz einfach. Wenn Sie sich nicht schon etwas mit Netzwerktechnik auskennen, kann die Fülle an Informationen Sie im ersten Moment erschlagen. Zumal es sich der Autor nicht nehmen lässt, immer wieder auf die Hintergründe und Zusammenhänge einzugehen – auch wenn diese etwas weiterführen, als es für das Projekt unmittelbar notwendig wäre.

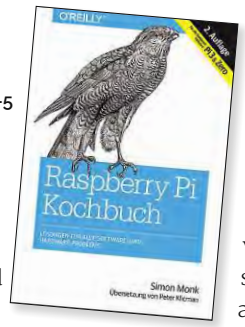
Wenn Sie sich jedoch darauf einlassen, können Sie lernen, wie Sie Ihren gesamten Wohnbereich mit dem Raspberry Pi verbinden und wie Sie am Ende Licht, Temperatur und mehr steuern – auch aus der Ferne wie zum Beispiel aus dem Urlaub. Wer spielerische, anschauliche Schritt-für-Schritt-Anleitungen sucht, sollte von dem Buch jedoch besser die Finger lassen.

Klasse: Die teilweise sehr langen Codelistings können Sie unter **buch.cd** herunterladen oder sich als CD zusenden lassen.

Fazit ★★★★☆

RASPBERRY PI KOCHBUCH

Autor: Simon Monk
Verlag: O'Reilly
Preis: €29,90
ISBN: 978-3-96009-033-5
Info: bit.ly/2leUWtK



Das Raspberry Pi Kochbuch ist gerade in der zweiten Ausgabe erschienen und für den Raspberry Pi 3 und den Zero überarbeitet worden. Das Buch hat den Anspruch, für alle Software- und Hardware-Probleme mit dem Raspberry Pi eine Lösung parat zu haben. Auch wenn das für manchen einer Herausforderung gleichkommen mag: Die Suche nach einem Problem, das sich mit dem Buch nicht lösen lässt, ist zumindest anspruchsvoll.

Die Bandbreite reicht von sehr einfachen Themen wie „Eine MicroSD mit Noobs beschreiben" oder „Kabelgebundener Anschluss an ein Netzwerk" bis hin zu Charlieplexing oder „Temperatur mit Thermistor messen". Ein Kapitel ist dem Thema Arduino und Raspberry Pi gewidmet. Angereichert ist das Ganze mit Schaltplänen und Codelistings. Außerdem finden Sie umfangreiche Listen von Einzelteilen sowie Herstellern und Bezugsquellen für allerlei Zubehör und diverse Extras im Anhang.

Das Buch richtet sich dabei aufgrund seiner Sprache nicht an blutige Anfänger. Das gängige Vokabular sollte man schon kennen. So wird zum Beispiel von Anfang an vorausgesetzt, dass man weiß, was GPIO-Pins sind. Erfreulich ist bei der großen Fülle der abgedeckten Themen neben dem klar strukturierten Inhaltsverzeichnis auch das ausführliche Glossar.

Fazit ★★★★★

MIT DEM RASPBERRY PI ZUM EIGENEN HOMESERVER

Autor: Bernhard Münkel
Verlag: Tredition
Preis: € 14,99
ISBN: 978-3-7345-6739-1
Info: bit.ly/2lKV0zf

Egal ob Sie einen Mediaserver für Ihre Filme und Serien oder einen Webserver für Ihren Wordpress-Blog aufsetzen möchten – in diesem Buch werden Sie fündig. Bernhard Münkel erklärt das Thema ausführlich, Schritt für Schritt und von allen Seiten. Er stellt die Basisdienste vor und erklärt auch, worauf Sie bei der Datensicherung in Bezug auf die SD-Karte Rücksicht nehmen müssen. So lernen Sie nebenbei den Raspberry Pi und seine Vor- und Nachteile als Homeserver im Detail kennen. Uns haben besonders gut das Kapitel zum eigenen Radiosender und die Anleitung zu Owncloud gefallen. Mit Owncloud richten Sie nicht nur einen Cloudspeicher ein, sondern können auch Ihre Termine und Kontakte systemübergreifend verwalten. Zusätzliche Hilfen sind ein Kapitel zu praktischen Linux-Kommandos sowie zahlreiche Links unter jedem Abschnitt, über die Sie sich in die Themen weiter einlesen können. Auf ein Glossar wurde verzichtet, doch das ist angesichts des sehr ausführlichen Inhaltsverzeichnisses zu verschmerzen. Den Stern Abzug geben wir, weil die Screenshots in der Printausgabe teilweise extrem schlecht zu lesen sind.

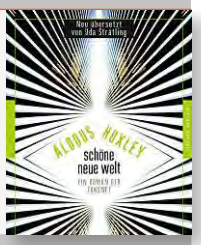
Fazit ★★★★☆

ROMAN-EMPFEHLUNGEN
SCI-FI KLASSIKER

Entdecken Sie ferne Welten und durchleben Sie düstere, dystopische Zukunftsvisionen

Schöne Neue Welt

Autor: Aldous Huxley
Herausgeber: Fischer
Preis: € 9,99
ISBN: 978-3-596-90573-7
Info: bit.ly/2aQCHYl



Eines der größten utopischen Bücher des 20. Jahrhunderts (1932). Die Neuübersetzung ist von 2014.

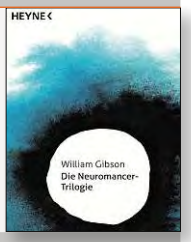
Die Foundation-Trilogie

Autor: Isaac Asimov
Herausgeber: Heyne
Preis: € 16,99
ISBN: 978-3-453-52795-9
Info: bit.ly/2kRRiWh

Obwohl schon 70 Jahre alt, sind die Zukunftsprognosen sowie die epische Vision Asimovs immer noch beeindruckend. Ein absolut lesenswerter Klassiker.

Neuromancer

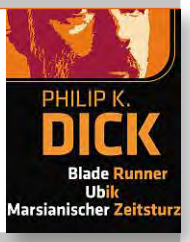
Autor: William Gibson
Herausgeber: Heyne
Preis: € 9,99
ISBN: 978-3-453-31598-3
Info: bit.ly/2lOALRr



Der Kultroman der 80er ist Vater eines ganzes Genres, des Cyberpunks. Er hat Film und Fernsehen geprägt.

Blade Runner

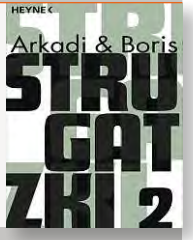
Autor: Philip K. Dick
Herausgeber: Heyne
Preis: € 14,00
ISBN: 978-3-453-52583-2
Info: bit.ly/2l7bMsD



Ursprünglich unter dem Titel „Träumen Androiden von elektrischen Schafen?" erschienen. Der Film von 1982 weicht erheblich von der Buchvorlage ab.

Picknick am Wegesrand

Autoren: Arkadi und Boris Strugazki
Herausgeber: Heyne
Preis: € 12,99
ISBN: 978-3-453-52631-0
Info: bit.ly/2kwAVuS



Metro 2033, die Computerspiele im selben Universum und S.T.A.L.K.E.R. greifen Elemente dieses russischen Klassikers um eine gefährliche Schatzsuche auf.

COMMUNITY-PROFIL

JILLIAN OGLE

Herrin über abenteuerliche Roboter, Pokémon-Fänger und Livestream-Brillen

Jillian Ogle

Gründerin und CEO von Let's Robot

Job: Erfinderin

Webseite: letsrobot.tv, magpi.cc/2gEZp3S

Let's Robot ist ein Videochannel von Jillian Ogle, der über die Webseite **letsrobot.tv** oder direkt über Twitch angeschaut werden kann. Regelmäßig Dienstags und Donnerstags können hier die Zuschauer über die Chatfunktion zusammen Roboter steuern. Dazu gibt es eine Liste von festgelegten Befehlen. Die Zuschauer haben die Möglichkeit, diese Befehle in den Chat einzugeben und darüber die Roboter zu kontrollieren. Damit das Ganze nicht in ein komplettes Chaos ausufert, gilt es immer wieder verschiedene Aufgaben zu erledigen:

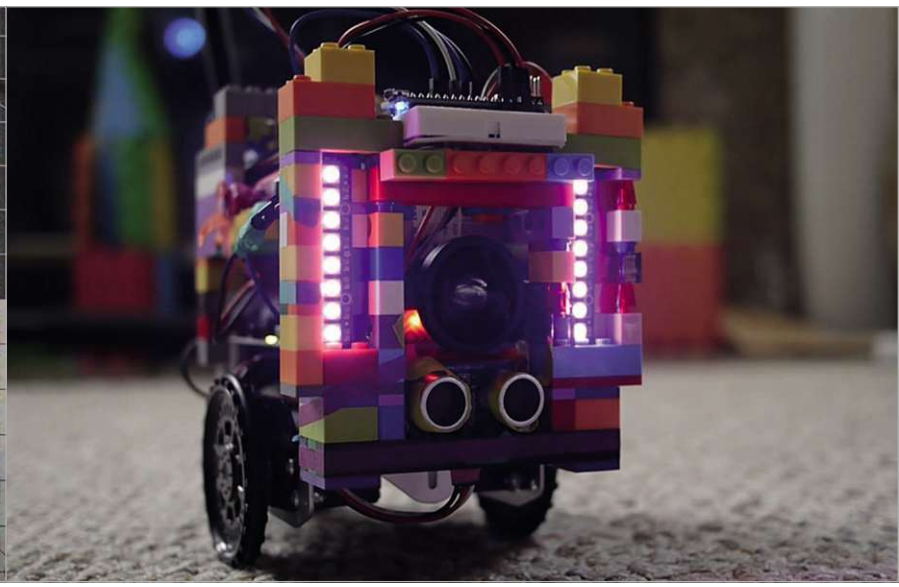
Mal muss ein Irrgarten durchfahren werden, mal werden Hinweise versteckt oder es müssen Gegner überwunden werden. Die Zuschauer auf Twitch.tv müssen zusammenarbeiten und Kommandos eintippen, um die Roboter durch das Terrain zu navigieren und die gestellte Aufgabe zu lösen.

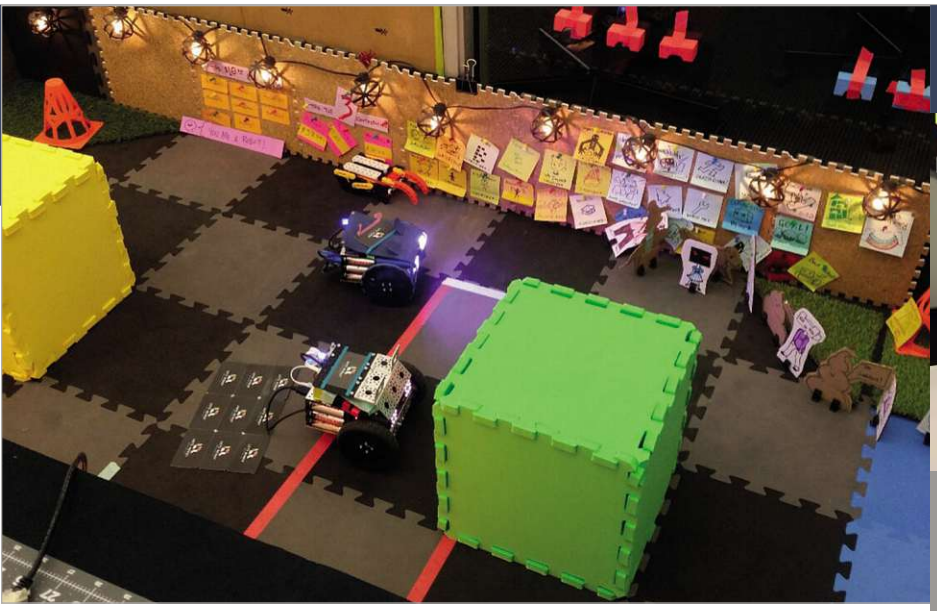
Der erste Roboter des Projekts war Aylobot. Sein Gestell besteht aus Lego-Steinen. Ninabot hingegen, eine Art Version 2.0 von ihm, hat einen Greifer und bietet so mehr Interaktionsmöglichkeiten. Beide sind mit einer Kamera ausgestattet, die zu Twitch streamt. So sehen die Zuschauer jederzeit das, was die Kameras einfangen. Inzwischen sind einige neue Roboter hinzugekommen – jeder hat seine speziellen Fähigkeiten.

Let's Robot ist eine Idee von Jillian Ogle –„die erste interaktive Live-Show der Welt, die Roboter

Unten Jillian: „Diese Post-Its habe ich für unsere #LetsRobot-Abonnenten gemalt. Sie sind Teil der Sets, durch die die Roboter navigiert werden"

Unten rechts Das Ziel von Let's Robot ist es, die Zuschauer aktiv einzubinden





Oben Die interaktiven Sets beanspruchen viel Raum im „Global Startup-Incubator The Batchery", Oakland, Kalifornien

einsetzt, die von den Zuschauern gemeinschaftlich gesteuert werden". Jillian wurde schnell klar, dass es die dafür nötigen Roboter noch nicht gab … und so entstand Let's Robot. Nachdem sie sich verschiedene mögliche Komponenten genauer angeschaut hatte, entschied sich Jillian für den Raspberry Pi. Seitdem steckt in jedem der Roboter ein Pi.

Vor Let's Robot arbeitete Jillian als Grafik- und Spieledesignerin.

> „Seit dem Start von Let's Robot steckt in jedem der Roboter ein Raspberry Pi"

Sie war Art Director von Playdom, einer Tochter von Disney Interactive, ging dort weg, um 2013 Aylo Games zu gründen und rief schließlich 2015 Let's Robot ins Leben. Damals kannte sie sich mit Hardware noch nicht gut aus. Sie hat sich das entsprechende Know-how daher in den letzten zwei Jahren selbst beigebracht.

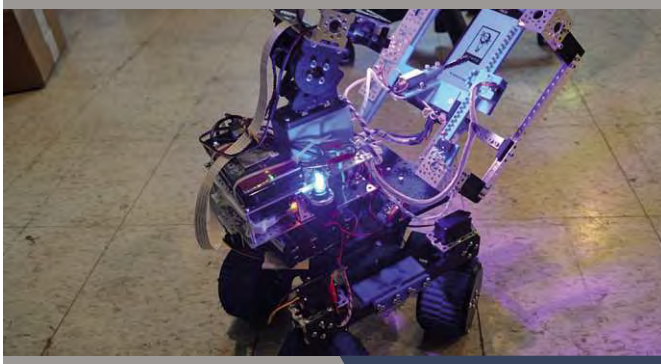
Nachdem die Abonnentenzahlen von Let's Robot weiter wachsen, ist Jillian jetzt immer mehr damit beschäftigt, das Projekt und die Roboter über Social Media bekannt zu machen. Neben den Robotern arbeitet Jillian aber noch an weiteren Projekten, die das Angebot von Let's Robot erweitern und verschiedene Trends im Bereich Social aus aller Welt aufgreifen. Ein Beispiel ist der Pokémon Go Roboter. Ursprünglich war das nur ein Greifarm, mit dem der Wurf eines Poké-Balls simuliert werden konnte. Später bekam er Räder und nun kann man mit ihm Pokémons in der freien Wildbahn jagen. Häufig trägt Jillian auch ihre selbstgebaute WiFi-Livestream-Brille in den Streams. Die Brille ist ebenfalls mit einer Pi-Kamera ausgestattet, damit die Zuschauer alles aus Jillians Sicht mitverfolgen können. Sehr spannend ist das jüngste Projekt: Derzeit arbeitet sie an einem Pi-betriebenen, transparenten Head-up-Display. Unter Berücksichtigung der lichtbrechenden Eigenschaften von Acryl will Jillian ein durchsichtiges Display bauen, auf dem sie die Nutzerkommentare im Twitch-Chat lesen kann, ohne auf einen anderen Monitor schauen zu müssen.

HIGHLIGHTS

twitch.tv/letsrobot

LET'S ROBOT

Zweimal pro Woche können die Zuschauer die Roboter selber online lenken. Dabei gilt es, Aufgaben zu lösen: zum Beispiel den Praktikanten zu füttern, Einladungen für eine Party zu schreiben oder einen Gegner zu besiegen.



magpi.cc/2hg0ovz

POKÉMON GO ROBOTER

Zum Beginn war der Pokémon Go Roboter an den Schreibtisch gefesselt – bis er Räder bekam. Nun jagt er draußen Pokémons und kann auch unwegsames Gelände bewältigen.

magpi.cc/2hfWe6T

LIVESTREAM-BRILLE

Mit Hilfe der WiFi-fähigen Livestream-Brille können die Zuschauer alles verfolgen, was Jillian sieht. Dafür wurde eine Raspberry-Pi-Kamera zwischen den Augen befestigt. Als Gag hat Jillian LED-Augenbrauen designt, die ebenfalls übers Internet gesteuert werden.

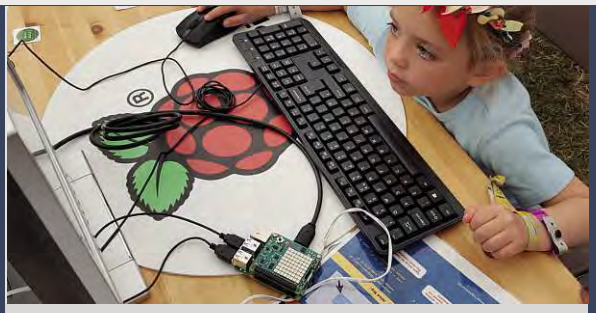
DIE WELT DES RASPBERRY PI

Wir zeigen Ihnen, was rund um den Mini-PC alles geplant ist

EHRGEIZIGE ZIELE FÜR 2017

Dieses Jahr feiert der Raspberry Pi seinen fünften Geburtstag. Der Pi war in den letzten Jahren sehr erfolgreich, doch das soll noch lange nicht alles sein. Philip Colligan, CEO der Raspberry Pi Foundation, hat in einem Blogeintrag auf der Homepage die Pläne der Stiftung für 2017 vorgestellt (**magpi.cc/2ijU9rn**). Zu Beginn geht er dort jedoch noch eimal auf die vielen Erfolge im vergangenen Jahr ein: „Die Raspberry Pi Foundation hatte ein fantastisches Jahr. Wir haben 2016 mit über 11 Millionen verkauften Raspberry Pis abgeschlossen. Millionen Menschen nutzten unsere Lernressourcen. Fast 1.000 Pädagogen wurden in Großbritannien und in den USA zertifiziert. 75.000 Kinder nahmen regelmäßig an über 5.000 Code-Clubs in Großbritannien teil. Hunderte von Raspberry-Jams fanden in der ganzen Welt statt. Von Schulkindern geschriebener Code wurde im All (ja, im Weltall) ausgeführt, und vieles mehr."

Hier sind einige Punkte, die 2017 laut Philip für den Pi und die Foundation von Bedeutung sind:



NEUE JUNGE DIGITALE MAKER

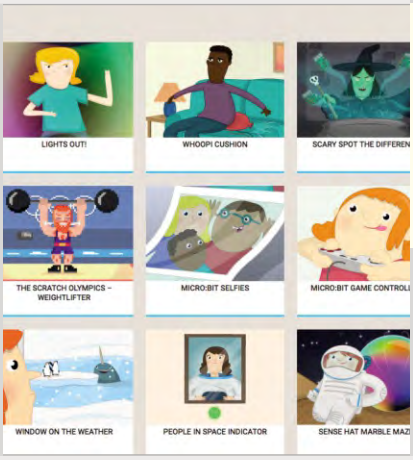
Es war von Anfang eines der Ziele der Raspberry Pi Foundation, gerade Kinder und Jugendliche für Computer und Programmieren zu begeistern. 2017 werden weltweit neue Code-Clubs entstehen und es gibt einen neuen Wettbewerb namens „Pioneers".

DIE COMMUNITY STÄRKEN

Die rege Community ist extrem wichtig für den Raspberry Pi – sie ist sogar ein bedeutendes Verkaufsargument. Die Stiftung wird deshalb ihren Wirkungsbereich erweitern, und wird eine Wohltätigkeitsorganisation in den USA gründen.

KOSTENLOSE LERNRESSOURCEN

Die große Zahl kostenloser Online-Tutorials und Pläne für Unterrichtsstunden etwa zum Programmieren auf dem Pi wird 2017 weiter ausgebaut. Die neuen Inhalte sollen Lehrer im Unterricht unterstützen. Darunter werden auch viele Videos sein. Alle Lerninhalte sind allerdings derzeit leider nur in Englisch verfügbar.



LEHRER UNTERSTÜTZEN

Neben der weiter verfügbaren Zertifizierung von Pädagogen durch die Picademy-Kurse gibt es eine brandneue Reihe von Online-Kursen. Weitere Informationen finden Sie unter **raspberrypi.org/education**.

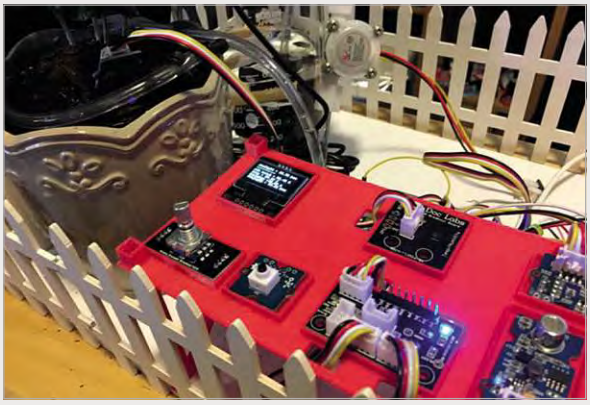
NEUE PROJEKTE

Hier stellen wir Ihnen die besten Crowdfunding-Ideen vor

AUDIO INJECTOR OCTO SURROUND SOUND

kck.st/2i5frsO

Es gibt einige Audio-Karten für den Raspberry Pi, aber diese ist etwas ganz Besonderes. Das Team von Audio Injector hat einen Surround-Sound-HAT entwickelt, wobei sowohl die Ein- als auch die Ausgänge mittels Cinch verbunden werden. Unser erster Gedanke war, dass es sich perfekt für ein System mit Kodi eignen könnte. Doch tatsächlich gibt es auch viele andere Audio-Projekte, für die sich dieser HAT einsetzen lässt.



SMARTPLANTPI

kck.st/2idtpsG

Vielleicht kennen Sie schon den FarmBot, den Landwirtschafts-Roboter? SmartPlantPi ist quasi eine Mini-Version davon, die sich um Ihre Topfpflanze kümmert. SmartPlantPi versorgt die Pflanze automatisch mit Wasser und überwacht zusätzlich mehrere Unweltfaktoren. So brauchen Sie demnächst nicht mehr Ihre Nachbarn zu bitten, im Urlaub Ihre Pflanzen zu gießen. Das Projekt eignet sich auch sehr gut für Kinder.

AUSSERDEM ...

Weitere Projekte, die wir spannend finden

magpi.cc/2j4RUFx

PI BRÜHT KAFFEE

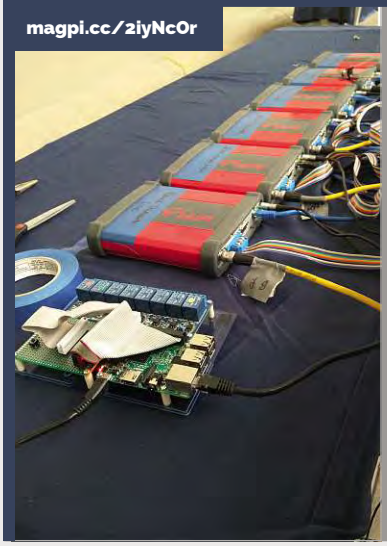
„Habe eine kleine, coole Anwendung für meine @wilburcurtisco Kaffeemaschine im Büro geschrieben", schreibt matty.mcc auf Instagram. Auf dem LC-Display, das mit einem Raspberry Pi verbunden ist, kann man sehen, wann die letzte Tasse gebrüht wurde und man bekommt jetzt eine E-Mail, wenn der Kaffee fertig ist.

magpi.cc/2j51Yyk

BMO

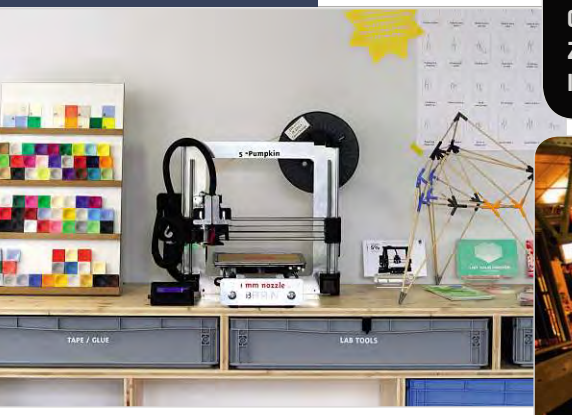
Wer hätte nicht gern einen Gamer-Kumpel wie den BMO aus Adventure Time auf Cartoon Network. NinjaBunny9000 hat sich einen gebaut. Das Herzstück ist natürlich ein Raspberry Pi, plus sehr viel maßgeschneiderte Elektronik. Das Gehäuse hat NinjaBunny9000 mit einem 3D-Drucker erstellt.

magpi.cc/2iyNcOr



TRAINING FÜR OLYMPIA

Das Olympic Training Center Velodrome in Colorado Springs, USA, scheint eine spezielle Konfiguration mit dem Raspberry Pi und RC-Decodern von MyLabs zu nutzen, um die Bahnen und Zeiten der Athleten aufzuzeichnen.



VERANSTALTUNG:
Pi Mini Robot (Berlin März Edition)
ORT: FabLab Berlin
ZEIT: 25. und 26. März 2017
INFO: bit.ly/2lRQqzz

VERANSTALTUNG:
Maker Faire Ruhr
ORT: DASA, Dortmund
ZEIT: 25. und 26. März 2017
INFO: makerfaire-ruhr.com

VERANSTALTUNG:
Make Munich
ORT: Zenit, München
ZEIT: 06. und 07. Mai 2017
INFO: make-munich.de

VERANSTALTUNGSKALENDER

>RASPBERRY PI BIG BIRTHDAY WEEKEND
04 | 03 | 2017
Mit großer PiParty, vom 4. bis zum 5. März in Cambridge.
Unter 16 Jahren Eintritt frei

>MINECRAFT HACKING
11 | 03 | 2017
Am 11. und 18.März im Erfindergarden in München

>MAKE RHEIN-MAIN
17 | 03 | 2017
Vom 17. bis zum 19. März in den Hallen unter der Honsellbrücke, Frankfurt

>MINECRAFT HACKING
18 | 03 | 2017
Am 18. und 19. März in den Sektor5 Coworking Spaces, Wien

>PI PRINGELS ROBOT
18 | 03 | 2017
Baue deinen eigenen Roboter am 18. und am 19. März in den Sektor5 Coworking Spaces, Wien

>MINI MAKER FAIRE
19 | 03 | 2017
Parallel zur Amateurfunk-Ausstellung SAFA in der Stadthalle in Dillingen, Saar

>CEBIT
20 | 03 | 2017
Vom 20. bis zum 24. März auf dem Messegelände in Hannover

>MAKER FAIRE RUHR
25 | 03 | 2017
Am 25. und 26. März in der DASA Arbeitswelt Ausstellung in Dortmund

>MINECRAFT HACKING
25 | 03 | 2017
Am 25. und 26.März im FabLab Berlin sowie parallel im Erfindergarden in München

>PI MINI ROBOT
25 | 03 | 2017
Baue deinen eigenen Roboter am 25. und am 26. März im FabLab Berlin

>BAU EINER LED-UHR
25 | 03 | 2017
Bastel-Workshop für Erwachsene im FabLab München

>PROJECTION MAPPING ADVANCED
25 | 03 | 2017
Am 25. und 26. März im Spektrum in Berlin. Kreieren Sie kompakte Projection-Mapping-Installationen mit dem Raspberry Pi und openFrameworks

>PROGRAMMIEREN LERNEN MIT DEM PI
25 | 03 | 2017
Für Kinder ab 10 Jahren. Am 25. März in den Coworking Spaces im Gewächshaus Düsseldorf

>PIWARS 2017
01 | 04 | 2017
Am 1. und 2. April in Cambridge, UK. Besuchertickets ab sofort verfügbar

>CODINGFERIEN OSTERN
10 | 04 | 2017
und 18 | 04 | 2017
Vom 10. bis zum 13. April und vom 18. bis zum 21. April in den Coworking Spaces Gewächshaus Düsseldorf

>MAKER FAIRE SACHSEN
22 | 04 | 2017
Am 22. und 23. April in der Stadthalle Chemnitz

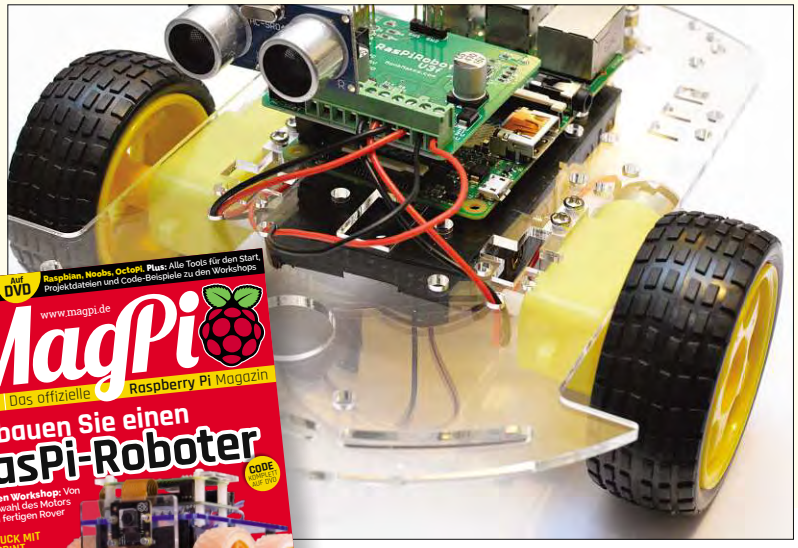
>NERDEND 2017
29 | 04 | 2017
Am 29. April im FabLab Würzburg

>MAKE MUNICH
06 | 05 | 2017
Am 6. und 7. Mai im Zenith – die Kunsthalle in München

>MAKER FAIRE VIENNA
20 | 05 | 2017
Am 20. und 21. Mai in der METAStadt in Wien

Fotos v. o.: DASA/Andreas Wahlbrink; FabLab Berlin; Tom Garrecht; MakeMunich/MakeGermany/Jenny-Ludwig Projekt

Aus: MagPi 1/2017, Seite 16



Bestellungen aus UK

Ich habe vor, mir einen Roboter nach der Anleitung aus dem MagPi 1/2017 zu bauen. Nun sind einige Teile nur in UK erhältlich. Beim ZeroBorg kostet der Versand beispielsweise 4,95 Pfund [rund 5,80 Euro]. Wird bei so etwas auch eine Zollgebühr fällig?
Adrian Lehwald

Keine Sorge, noch sind die Briten in der EU. Also fallen keine Zollgebühren an. Das kann sich allerdings ändern, wenn sie ausgetreten sind. An diesem Beispiel kann man einmal sehen, welche – unerfreulichen – Folgen der sogenannte Brexit auch im Kleinen haben kann.

Roboter im Eigenbau

Ich bin 13 Jahre alt und habe schon seit einem Jahr einen Raspberry Pi. In der Titelgeschichte der letzten MagPi 1/2017 ging es darum, wie man mit dem Pi einen Roboter baut. Dazu habe ich ein paar Fragen:
1. Auf Seite 29 wird beschrieben, wie man den Rover per Kamera zum Beispiel über ein Smartphone steuert. Dazu muss man den Roboter natürlich mit dem WLAN verbinden. Doch wie soll das funktionieren, wenn ich einen Pi Zero verwende? Der hat ja schließlich weder LAN noch WLAN an Bord.
2. Ich würde gerne einen Roboter mit Ketten bauen. Muss ich da irgendetwas an der Software verändern? Wenn ja, was? Ich muss zugeben, dass die ganze Thematik neu für mich ist.
Ach ja: Gibt es eine Möglichkeit, die Heft-DVD auch unter Mac OS zu öffnen?
Franz-Leopold Saitz

Du hast völlig recht: Will man etwa einen Rover per Webbrowser steuern, benötigt man entweder einen Raspberry Pi 3 mit integriertem WLAN-Modul oder muss einen separaten WLAN-Adapter anschließen.
Beim Pi Zero benötigt man hierfür noch einen USB-Adapter (Micro-USB auf Standard-USB). Da der Pi Zero nur einen USB-Anschluss hat, ist die WLAN-Einrichtung etwas kompliziert. Deutschsprachige Anleitungen findest du im Web, beispielsweise hier:
elektronik-labor.de/Raspberry/RpiZero.html
raspberry.tips/raspberrypi-tutorials/raspberry-pi-zero-wifi-einrichten
Aber dies ist ein Thema für sich und hätte den Rahmen unserer Roboter-Strecke im Heft einfach gesprengt.

Zu deiner zweiten Frage: Prinzipiell sind alle möglichen Roboter denkbar – auch solche mit Kettenantrieb. Unsere Titelgeschichte aus dem letzten Heft bietet einen Einstieg in ein schier unerschöpfliches Thema. Unser Vorschlag für den Beginn wäre jedoch erst einmal ein Roboter-Bausatz mit zwei Rädern. Diese Bausätze enthalten bereits meist alles, was man benötigt, und werden mit einer ausführlichen (leider in der Regel nur englischsprachigen) Anleitung ausgeliefert. Das ist ideal, um ins Thema Robotik einzusteigen und sich mit der grundlegenden Programmierung vertraut zu machen. Ein bekannter Bausatz ist beispielsweise der GoPiGo (ca. 100 Euro, **dexterindustries.com**). Zum GoPiGo gibt es zudem zahlreiche Tutorials im Web. In dieser Ausgabe des MagPi (siehe Seite 102) stellen wir auch eine günstigere Alternative vor: Das RasPiRobot Rover Kit von Monk Makes (ca. 50 Euro, **monkmakes.com**).
Übrigens: Alle unsere Heft-DVDs laufen unter Windows, Linux und auch Mac OS. Mit der DVD dieser Ausgabe lässt sich sogar Debian mit Pixel auf einem Mac starten.

SCHREIBEN SIE UNS

Sie möchten uns etwas zur MagPi mitteilen?
Kontaktieren Sie die Redaktion via **specials@chip.de**

MATT RICHARDSON

Matt fungiert für die USA als Raspberry-Pi-„Evangelist". Davor hat er am Buch *Raspberry Pi für Einsteiger* mitgeschrieben und war Autor für das Magazin „Make".

DIE NÄCHSTE AUSGABE VON MagPi ERSCHEINT AM 10. MAI 2017

LEBEN UND LERNEN

Das Leben ist ein permanenter Lernprozess. Und das ist auch gut so, findet **Matt Richardson**

Die Raspberry Pi Foundation hat bekanntermaßen einen Bildungsauftrag – und diesen verbindet man vielleicht erst einmal mit jungen Menschen: Schülern, die lernen, wie man programmiert, wie Computer funktionieren und wie man diese einsetzt, um sich zum Maker zu entwickeln. Oder man denkt an Lehrer, die die Ressourcen der Foundation nutzen, um ihre Schüler für das Basteln am Computer zu begeistern.

Wir alle wissen jedoch, dass das Lernen nicht im Klassenzimmer aufhört. Es findet auch zu Hause statt, in Bibliotheken, in Coding-Clubs, in Museen und an vielen anderen Orten. Wir als Mitarbeiter der Raspberry Pi Foundation geben uns große Mühe, junge Menschen an all diesen Orten abzuholen. Das ist auch der Grund dafür, dass viele unserer Dozenten nicht an Schulen unterrichten, sondern häufig auch in anderen Lernumfeldern tätig sind.

Obwohl es natürlich eine unserer Hauptaufgaben ist, junge Menschen weiterzubilden, ob im Klassenzimmer oder außerhalb: Wir kennen keine Altersgrenzen. Ob jung oder alt – wir freuen uns über jeden Menschen, der einen Pi oder unsere Lernplattformen nutzt. Wir wissen, dass unsere Community auch aus vielen Menschen besteht, die schon mitten im Leben stehen. Sie eignen sich Know-how an, experimentieren mit dem Pi oder betätigen sich als Maker.

Auch ich gehöre zu dieser Community von Menschen, die ihr Leben lang weiterlernen möchten. Seit ich den Raspberry Pi im Jahr 2012 das erste Mal in der Hand hatte, habe ich unendlich viel dazugelernt, indem ich mich zum Maker entwickelt habe.

Ursprünglich war es ehrlich gesagt nicht meine Absicht, mich intensiv mit dem Programmieren und Algorithmen auseinanderzusetzen. Aber ich tat es dann doch, quasi nebenbei, indem ich mich mit Projekten beschäftigte, bei denen entsprechendes Know-how gefragt war. Das ging sogar über das reine IT-Wissen hinaus. Beispielsweise musste ich mir selbst einen Mathe-Auffrischungskurs geben, als ich an meinem Projekt „Dynamisches Fahrradlicht" arbeitete. Ziel war es, auf Basis des Reifendurchmessers und der Umdrehungen pro Minute die Geschwindigkeit meines Fahrrads auszurechnen.

Ich gehe stark davon aus, dass es viele Menschen gibt, denen es so ähnlich geht wie mir: Sie basteln an ihren Pis herum und eignen sich nebenbei jede Menge Know-how an.

Auch wenn Sie auf einen Workshop zurückgreifen, um einen Retro-Arcade-Automaten zu bauen, einen Home-Server aufzusetzen oder einen magischen Spiegel zu entwerfen: Immer lernen Sie jede Menge dazu. Gute Tutorials bringen Ihnen nämlich nicht nur nahe, welche Schräubchen Sie drehen müssen, sondern erläutern auch, warum Sie das tun. Im besten Fall lässt Ihnen der Workshop dabei genügend Raum, um selbst zu experimentieren.

Vielleicht sind Sie auch schon im professionelleren IT-Umfeld unterwegs und möchten etwa eine Sandbox einrichten, um Server, Netzwerke, Cluster Computing oder Ähnliches zu testen? Dafür sind Raspberry Pis sehr gut geeignet, da sie so günstig und weit verbreitet sind. Letzteres hat den Vorteil, dass Sie, egal was Sie vorhaben, vermutlich Menschen finden, die damit bereits Erfahrung haben und Ihnen weiterhelfen.

Um zu den jungen Menschen zurückzukommen: Es ist wichtig, ihnen zu zeigen, dass wir, die Erwachsenen, das Wissen nicht gepachtet haben. Auch wir lernen weiter, manchmal mit den Jungen zusammen, manchmal sogar von ihnen. Und das ist das Wichtigste: Junge Menschen müssen begreifen, dass Lernen ein lebenslanger Prozess ist.

CHIP KIOSK

Im CHIP Kiosk finden Sie alle Produkte aus dem Hause CHIP. Magazine und Specials – gedruckt oder digital. Als Einzelheft oder Abo.

www.chip-kiosk.de

Das Hardware-Magazin für PC-Gamer

148 Seiten
Kaufberatung, Tests, Praxis

Nur 9,95 €

Jetzt bestellen.
www.chip-kiosk.de

reichelt elektronik
Technik verbindet.

Jetzt Newsletter abonnieren!
Unter allen neuen Newsletter-Abonnenten wird monatlich ein **technisches Highlight** verlost!
http://**rch.lt/v3**

reichelt.de @reichelt_el google.com/+reichelt

Foto: Foto- und Bilderwerk

„Der einfachste Weg den Raspberry Pi zu verwenden! Vielfältig modulierbar — das rechnet sich!"

pi-top

SO WIRD IHR RASPBERRY PI ZUM DESKTOP-COMPUTER

Einfach die vordere Abdeckung des pi-topCEED öffnen, Ihren RaspberryPi anschließen und mit vohandener Maus und Tastatur ist Desktop-Computer sofort einsatzbereit — dank vorinstallierter Software!

- 14"-HD-Bildschirm (1366 x 768 Pixel)
- verstellbarer, rutschfester Ständer
- Einsatz von Zusatzplatinen über die modulare Schiene ist möglich
- optimiert für den Raspberry Pi 3 (nicht im Lieferumfang enthalten)

RASP CEED PRO GR
169,90

Auch in grün erhältlich:
RASP CEED PRO GN 169,90

DER MAKER LAPTOP

pi-top

Der **pi-top** ist ein DIY-"Raspberry Pi"-Laptop. Schließen Sie einfach Ihren vorhandenen Raspberry Pi an und legen Sie sofort los! Ideal um unterwegs zu programmieren.

- 13,3"-HD-Bildschirm
- vorinstalliertes pi-topOS, auf SD-Karte (8 GB)
- Akkulaufzeit: 10 Stunden

RASP PI-TOP D GN
299,90

Auch in anthrazit erhältlich:
RASP PI-TOP D GR 299,90

10-MAL SCHNELLER ALS DIE 1. GENERATION

DIE NEUSTE GENERATION!
RASPBERRY PI 3

- 1200 MHZ Quad-Core-Prozessor ARM Cortex-A53 64-Bit
- 1024 MB RAM LPDDR2
- VideoCore IV Dual-Core-GPU
- WLAN, BT, LAN, USB, HDMI, 40 I/O, ...
- unterstützt Windows 10 IoT, Raspbian, RaspBMC, Arch Linux, ...

RASPBERRY PI 3
37,50 statt 39,90

Für pi-top Laptop/CEED:
ERWEITERUNGSPLATINE

- Prototypenentwicklung von elektr. Funktionen
- inkl. Montagematerial

RPI PI-TOP PROTO **9,70**

Für pi-top Laptop/CEED:
2-WATT-LAUTSPRECHER

- modularer Aufbau, bis zu 3 Speaker in Reihe möglich

RPI PI-TOP SPEAK **27,30**

GLEICH ONLINE SHOPPEN!
Alle Artikel dieser Anzeige finden Sie hier: http://**rch.lt/zF**

Tagespreise · Preisstand: 14. 2. 2017, Preise in € inkl. gesetzl. MwSt., zzgl. Versandkosten, reichelt elektronik, Elektronikring 1, 26452 Sande (D)

- über 45 Jahre Erfahrung
- schneller 24-Std.-Versand
- über 60.000 Produkte am Lager
- kein Mindermengenzuschlag
- zertifiziert nach DIN EN ISO 9001:2008 · ISO 14001:2015

www.reichelt.de
BESTELLHOTLINE: +49 (0)4422 955-333